Quark Shell Model Calculations of Baryon Interactions

Luke Taylor.

Department of Physics and Astronomy,
University of Glasgow.

Thesis submitted to the University of Glasgow
for the degree of Ph.D.
September 1995.

To my Grandad, Alec McLennan.

Abstract

In this thesis, the form of baryon-baryon interactions is investigated using a non-relativistic quark potential model, within the framework of the shell model. It describes the further development of the previous quark model studies carried out by the group. The two nucleon (or baryon) system is treated as a system of six quarks, split into two clusters. A two-body interaction potential based on gluon exchange between individual quarks is introduced, along with a phenomenological confinement potential. The energy of the system is calculated for different values of the cluster separation, to build up a picture of the overall interaction. The model is then extended to include $6q(q\bar{q})$ configurations in the model space, as a crude approximation to describing the meson excitations which are essential to a full description of the interaction. An appropriate potential is included to describe these $q\bar{q}$ excitation effects.

Chapter 1 provides an introduction to the nucleon-nucleon interaction and the quark model in general, and describes non-relativistic quark potential models. Chapter 2 gives background information on the shell-model and its applications and goes on to describe how such models from nuclear physics have been carried over into quark model calculations. It also gives more details of previous work in the area. In chapter 3, the theory behind the model and the details of its implementation are discussed in detail. Chapter 4 describes the gluon exchange potentials used in the model and the calculation of the matrix elements associated with them. The results of the calculations and analysis of the wavefunctions are given in chapter 5. In chapter 6, we examine the results in more detail and end with a critique of the model as it currently stands.

Declaration

Except where specific reference is made to the work of others, this thesis has been composed by the author. It has not been accepted in any previous application for a degree. I further state that no part of this thesis has already been or currently is being submitted for any degree or qualification at any other university.

Luke Taylor.

Acknowledgments

Firstly I'd like to thank my supervisor, the legendary Dr. Sandy Watt for his constant cheerfulness and his general enthusiasm for everything. Thanks also to the head of our group, and Dean of Science, Professor Rex Whitehead for generally looking out for us and writing me impressive references.

Thanks to the SERC (now the EPSRC) for providing the funding for my project.

I'd like to thank everyone in the physics department who've made it such a friendly place in which to do a Ph.D. In particular, Brian "Baz" Ewins, for being such an excellent person to share an office with for three years. His occasional forays into the fascinating world of desktop mycology and horticulture were more than offset by his sense of humour, bad puns and the many investigations we undertook together into life, beer, the universe and everything.

Fellow postgraduates (and others), past and present: The ex-pat Gerry Lowe, Leïla "what a life" Ayat, Jack "I'm not sharing a bed with Gerry" Ireland, Gary "Embittered Cynicism" Benson, Carolyn "Rear Window" Sellar, Auntie Chrissie and her aura, Rebecca "somethink nice" Crawford, happy couples Douglas and Sharon, Saad and Betty, David "Enjoy" Keston and Susan "SUSY" Morrison, Andrew "Heavy Sigh" Lidsey, Gordon "Gary-II" Jenkins, the other Davids, Hugh, Arifa, Mary and Frook, Paul, Chris "Catalogue Man" Barton, Gillian and Terry, who was hardly here long enough to unpack his bags but seemed like a nice guy anyway.

Gioti "κινούμενος θάμνος" Petkaki, for late night cups of sleepytime-tea, and keeping the Vatrachos Blues at bay.

Wǒ de "Dà Jǐe" Xiǎo Zhéng, xièxie, nǐ de zhōngwén kè, Rong, xièxie, nǐ de yǒuyì.

On the family side, my Mum and Stevie, Dad and Jacki for their support and making sure I was always well fed.

Thanks also to Gordon and Helen Main (plus family) and Finbar for trips away and days on the hill, and to Uncle Ken for his Highland hospitality with a hint of Italian.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Nuclear Force and the Quark Model

The atomic nucleus is built up of protons and neutrons (or nucleons) and is held together by the nuclear force between them - the nucleon-nucleon (NN) interaction. In 1935, Yukawa put forward his meson theory of the nuclear force [3, 4]. In analogy with the theory of the Coulomb force, which arises from the exchange of photons between charged particles, he postulated the existence of a new particle, the π-meson, or pion and suggested that the nuclear force was a result of these particles being exchanged between nucleons. The main difference is that the photon is massless, giving rise to an infinite range force, whereas the pion has a mass and the nuclear force only acts over a short range. Yukawa showed that the force was of the order $e^{-\mu r}/r$ and he was able to obtain a reasonable estimate of the pion's mass, μ using estimates of the range of the nuclear force which was roughly known at the time.

Later, the existence of other, much heavier mesons such as the ρ and ω was also predicted. These are exchanged at short range and also have an effect on the NN interaction. The ω exchange is strongly repulsive and is thought to be one of the main ingredients of the strong repulsive core which is such a major feature of the interaction – see figure 1.1.

At the Stanford Linear Accelerator Center (SLAC) in the late 1960's, data from deep-inelastic electron-nucleon scattering experiments showed that nucleons are themselves composite particles. These were analogous to Ruther-

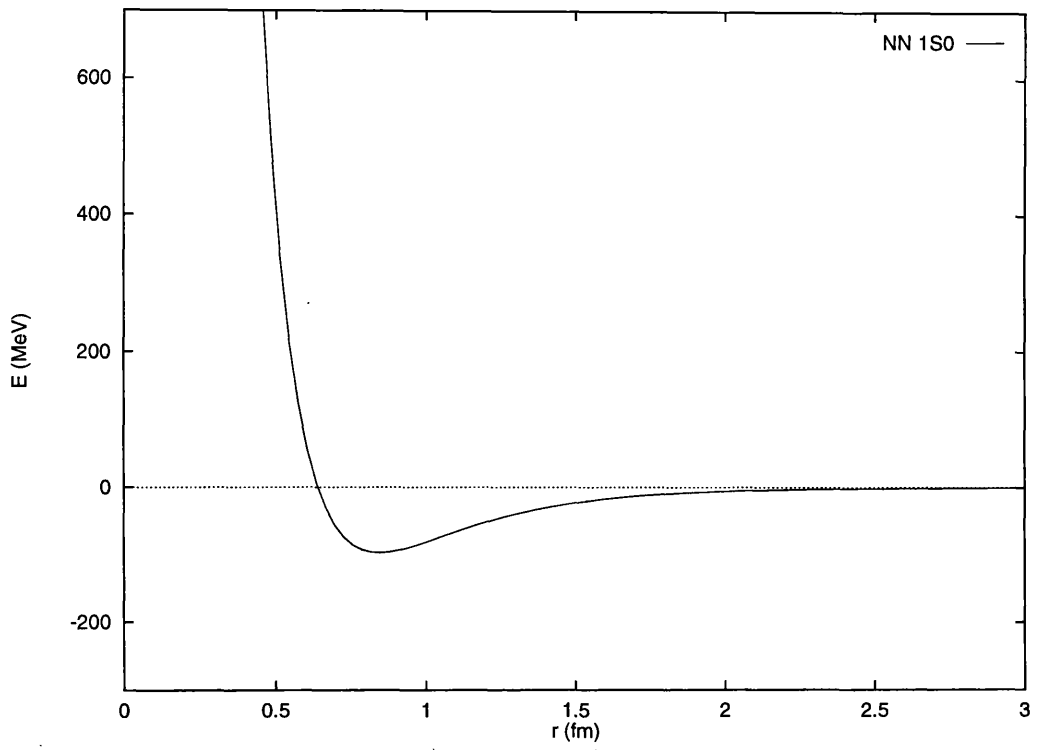

Figure 1.1: The Nucleon-Nucleon interaction ($^1S_0$ Channel)

ford's famous alpha-particle scattering experiments which revealed the existence of the atomic nucleus at the beginning of the century. Since then, various neutrino-nuclear scattering experiments have also provided evidence for the existence of point-like particles within the nucleon [5]. The discovery of these so-called *partons* [1] gave a new physical significance to the quark model of Gell-Mann and Zweig, which we discuss below.

1.2 Multiplets and Unitary Symmetry

Before the experimental evidence for partons was obtained, the known particles had been classified according to the symmetries they exhibited in their properties. On examining the spectrum of known particles, it was found that they could be grouped into families which had the same spin and parity, and almost the same mass, the only difference between the members of each fam-

[1] The name originally used by Richard Feynman for the point constituents of the nucleon

ily being their electrical charge. The classic example of this is the well known *isospin* symmetry — the proton and neutron form an isospin doublet and the three pions (π^-, π^0, π^+) an isospin triplet. Isospin was originally described using the isospin quantum number, T, which obeyed the same mathematics as spin. The nucleon has $T = \frac{1}{2}$ and comes in two forms: the proton (isospin up, $T_z = +\frac{1}{2}$) and the neutron (isospin down, $T_z = -\frac{1}{2}$). Similarly, the pion has $T = 1$, and T_z can take the values 1, 0 or -1.

Symmetries of this kind in particle physics can also be depicted using group theory. Specifically, isospin is associated with an underlying SU(2) group structure [2]. The multiplets of particles are then described by the representations of the group. The fundamental representation – called the 2 representation – of SU(2) is an object which can exist in two states, labelled up and down or (u, d). This describes the nucleons. The equivalent basic representation $(\bar{d}, \bar{u})$ describes the antinucleons. Other representations of the group can be formed by combining these basic representations — in particular, combining 2 and $\bar{2}$ gives a triplet (the pions) and also a singlet which is the η^0 meson [3].

In this way, SU(2) can be used to describe many of the hadrons using just the isospin quantum number. However, when strange particles were discovered, a group of larger dimensionality was needed and the use of SU(3) was suggested, giving rise to the *strangeness* quantum number.

The fundamental representation of SU(3) is three dimensional. As before, combinations of the 3 representation describe the multiplets of particles. Mesons are combinations of 3 and $\bar{3}$, producing a singlet and an octet (often just called a nonet). Examples of these are the familiar pseudoscalar and vector meson nonets. Baryons are a combination of three 3 representations which leads to baryon singlets, octets and decuplets.

This system of describing hadrons using SU(3) multiplets was useful in bringing some order to the large numbers of particles which had been discovered. It also led to the prediction of the existence of the Ω^- as well as its mass and quantum numbers from its position in the baryon decuplet.

[2] SU(2) – the *Special Unitary* group in 2-dimensions – is the group of rotational transformations in isospin (or real) space

[3] $\bar{2}$ is the conjugate representation $(\bar{d}, \bar{u})$.

1.3 The Quark Model

In 1964, Gell-Mann and Zweig both independently realized that the 3 representation could be thought of as describing a triplet of more fundamental constituent particles from which the known hadrons were constructed. Gell-Mann called these particles *quarks*. The three types, or *flavours*, of quark are labelled *up*, *down* and *strange* (u, d, s)[4].

In this picture, baryons are systems of three quarks (qqq), while mesons consist of a quark-antiquark pair $(q\bar{q})$. The quantum numbers of these new particles can be ascertained from those of the members of the baryon decuplet (details can be found in [5]). In particular to give each baryon the correct spin, the quarks must each have spin 1/2.

Although the idea of hadrons being constructed of quarks was not given much credence in the early days, this changed with the experiments mentioned above and the discovery that hadrons really did appear to be composite particles. Quarks are now seen as being the true constituents of mesons and baryons.

1.3.1 Further Flavours of Quark

The SU(3) quark model adequately describes the particles which were known at the time of its inception and is more than sufficient for the particles we shall be studying in this work. However, for completeness' sake, it should be noted that further experimental discoveries and advances in theory have led to the introduction of an additional three quark flavours and associated quantum numbers — c, b and t, for charm, bottom (or beauty) and top (or truth). The quantum numbers of the full six-quark family are given in table 1.1.

1.4 Coloured Quarks

As we have described it so far, the quark model is in fact incomplete. We can see this immediately by examining some of the members of the baryon

[4]These three flavours were all that was required to describe the known hadrons at the time. Later discoveries of new particles necessitated the introduction of additional flavours

	u	d	s	c	b	t
Charge, Q (units	2/3	-1/3	-1/3	+2/3	-1/3	+2/3
Isospin, T	1/2	1/2	0	0	0	0
T_z	1/2	-1/2	0	0	0	0
Strangeness, S	0	0	-1	0	0	0
Charm	0	0	0	1	0	0
Bottom	0	0	0	0	1	0
Top	0	0	0	0	0	1

Table 1.1: Properties of the different quark flavours

decuplet more closely. The Δ^-, Δ^{++} and Ω^- are made up of three S-wave d, u and s quarks respectively. Like all the members of the decuplet, they have spin 3/2, so to achieve this the three quarks must all have their spins aligned in parallel.

On theoretical grounds, we expect quarks to be fermions since baryons (consisting of 3 quarks) are fermions. Therefore they should obey the Pauli exclusion principle and it should not be possible for them to exist in the same state. So the three baryons in question would appear to be forbidden states.

An alternative explanation is that there is another hidden degree of freedom for the quarks, which must have at least three different values. If the quarks are in different states of this additional degree of freedom then the states are no longer Pauli forbidden.

This additional quantum number is called *colour* and the three possible colours are usually labelled red, green and blue, the primary colours of white light. This analogy stems from the fact that hadrons are colourless — only those in which the colour quantum numbers of the quarks cancel to give zero are observed. This picture also explains why only combinations of 3 quarks or quark-antiquark pairs produce physical bound states, since only these combinations can be colourless.

The significance of the colour quantum number goes much deeper than just satisfying the exclusion principle or explaining why quarks only bind in particular combinations. In fact, it is seen as being the true source of the strong interaction — the colour quantum number is the underlying "charge", playing

the same role as electric charge in the electromagnetic interaction. In the same way that the electromagnetic interaction is the exchange of photons emitted and absorbed by electric charge, the true strong interaction is the emission and absorption of bosons called *gluons* emitted and absorbed by colour charge. So the nuclear force which was once thought to be the fundamental force of the strong interaction is now seen as being due to the residual effects of the true strong force between quarks. A suitable analogy is the manner in which molecular forces between electrically neutral atoms arise as the remainder of the Coulomb force between charged objects.

There is also strong experimental evidence for the existence of colour. The results of electron-positron annihilation experiments are often quoted as an example, where the calculated value of

$$R = \frac{\sigma(e^+e^- \rightarrow hadrons)}{\sigma(e^+e^- \rightarrow \mu^+\mu-)} \tag{1.4.1}$$

is only consistent with experimental data if an extra factor of three is included to account for the colour degree of freedom. Details can be found in any good particle physics textbook, or see [6].

1.4.1 Quantum Chromodynamics

The theory of the strong interaction is called *Quantum Chromodynamics* (QCD), after the theory of *Quantum Electrodynamics*. The latter provides such an excellent description of the electromagnetic interaction that it was only natural that it should influence the search for the new theory. The most important features of QED that are taken over to QCD are gauge invariance and local gauge symmetry.

Like flavour, QCD is based on the idea that the three colours form the basis of an SU(3) group and it is hypothesised that only singlet representations of SU(3) colour are physically realised states. For states with three particles or less, only $q\bar{q}$ and qqq can form colour singlets, thus explaining why other combinations are not observed.

The gluons belong to an octet of SU(3) which means that they also carry colour. This is a major departure from the analogy with QED where the gauge boson which carries the force (the photon) does not have a charge. Gluons can

thus interact with each other via the same colour forces that act on coloured quarks and should also be able to form colour singlet states composed of combinations of gluons or a mixture of gluons and quarks.

Gluon-gluon interactions have no analogue in QED, and lead to properties of the strong interaction which differ markedly from those of the electromagnetic interaction.

In particular they are thought to be responsible for the property of *asymptotic freedom* whereby the strong coupling constant tends to zero as the momentum transfer of the scattering process increases — when looked at with very high energy probes, the quarks and gluons appear to be almost like free particles.

Traditionally in field theory, the calculation of quantities of physical interest, such as scattering amplitudes and matrix elements, requires the use of perturbation theory where a power series in the coupling constant is used. Higher terms in the series represent more complicated possibilities for exchange of virtual gauge bosons between fermions. Collectively, these differing processes are represented by Feynman diagrams. In the case of QED, the characteristic coupling constant (the fine structure constant) is small ($\approx 1/137$) and the series converges rapidly and is relatively easy to sum.

A related consequence of the self-coupling of gluons in QCD is that the interaction between quarks increases with their separation. Higher order processes will then become more important, making calculations using perturbation theory impossible. This so-called *infrared slavery* is thought to be responsible for the confinement of colour — the fact that we only observe colourless hadrons in nature and do not observe free quarks.

1.4.2 Quarkonium Potentials

A convenient illustration of these features of QCD can be found in the study of the heavy quark mesons charmonium ($c\bar{c}$) and bottomonium ($b\bar{b}$) which were first observed in the 1970's. Their discovery was a major triumph of the quark model since the existence of the charmed quark had been predicted as far back as 1967.

As more charmonium states were detected, it was observed that its spec-

trum bore a striking similarity to that of positronium (e^+e^-) which suggests that the potential describing the interactions might be of a similar form. The positronium spectrum can be described very well by solving the Schrödinger equation with a Coulomb potential and it was found that the energy levels of charmonium could be fitted using a "Coulomb plus linear" potential of the form

$$V_c(r) = -\frac{a}{r} + br \tag{1.4.2}$$

The first Coulomb-like term is due to one-gluon exchange, analogous to the $1/r$ potential arising from one-photon exchange in QED. The second term is negligible at short distances, in line with asymptotic freedom, but dominates as the particle separation increases, giving rise to a steadily increasing *confining potential* which prevents the particles from escaping.

Modern calculations use lattice gauge theory to give a proper interpretation of the mass of the nucleon and other hadrons. The hadron spectrum can now be obtained quite convincingly, but properties of multi-hadron systems such as the Deuteron, Triton and other nuclei are still beyond the scope of present-day calculations. Relativistic quark potential models have also been used. The simplest model however is the non-relativistic quark potential model in which the light quarks are given a mass of about 1/3 GeV. Although we believe that the quark masses are much less than 1/3 GeV and that this value is merely an effective mass, the model is surprisingly successful.

1.5 Non-Relativistic Quark Potential Models

> "Astonishingly, the nucleon's static properties appear to be quite successfully described by pretending that the nucleon is composed of three independent quarks each of 1/3 GeV".
>
> *Frank Close*

The quarkonium potentials discussed above work well for heavy quarks like c and b, but can potential models also be applied to light quarks systems such as the nucleon and other light baryons for which a non-relativistic treatment would seem to be difficult to justify? This might seem like a hopeless cause,

but, as the above quotation from Frank Close indicates, a convincing fit of the light baryon spectrum can be obtained using models of this kind. This is

Baryon	Theory	Experimental
N	940	940
Λ	1110	1115
Σ	1190	1195
Ξ	1325	1320
Δ	1240	1240
Σ^*	1390	1385
Ξ^*	1530	1535
Ω	1675	1670

Table 1.2: Comparison of non-relativistic quark model and experimental masses (in MeV) of ground-state baryons

illustrated in table 1.2, taken from [7]. The theoretical results agree very well with experimental values. As in the case of the quarkonium potential, colour confinement is achieved by means of a strong, confining potential which tends to infinity as the particle separation r tends to infinity. It is often expressed as an integer power of r. There is also a hyperfine interaction, as derived by

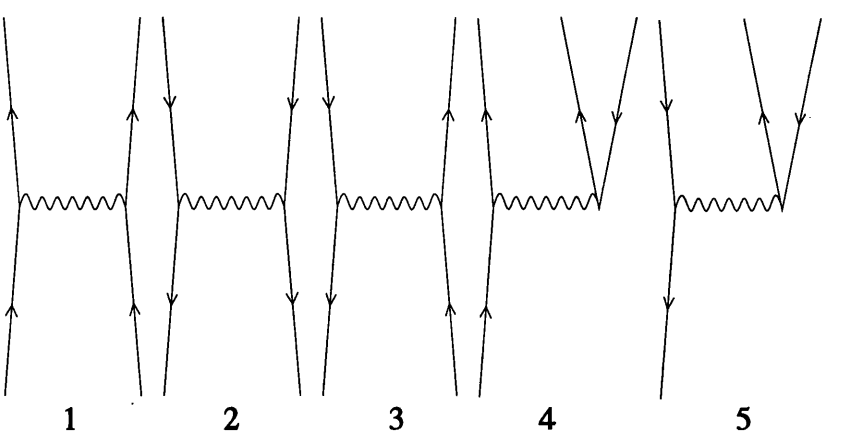

Figure 1.2: One-gluon exchange diagrams used in our model

De Rújula, Georgi and Glashow [8], which is a non-relativistic approximation

to the one-gluon exchange forces between two quarks. Its exact nature - the terms included and parameters used - varies between implementations.

This model can also be used for a calculation of the nucleon-nucleon interaction [9], which is the remnant of gluon exchange between two 3-quark clusters, just as the Van der Waal's forces between atoms are residues of the Coulomb interaction.

More details of the form of potential we use can be found in the discussion of potentials in chapter 4 — figure 1.2 shows Feynman diagrams for the terms relevant to our model. We include only one-gluon exchange terms for reasons given later. The first three terms are number conserving while the last two terms change the number of particles in the system. Term 1 in figure 1.2 represents gluon exchange between two quarks, term 2 represents exchange between two anti-quarks and term 3 represents exchange between a quark and an anti-quark. Term 4 (5) represents processes in which a gluon, generated from a quark (anti-quark), annihilates to give a quark-antiquark pair. The quark masses used tend to be about a third of the nucleon mass, in contrast with relativistic models, where they are just a few MeV. These, so called *constituent* masses, can be thought of as being due to the energies associated with confinement [10].

In early calculations of this kind, the 3 quarks were confined to the 0S shell and oscillator wavefunctions were often used. The next step was to try to calculate the spatial distributions of the baryons. An excited state requires a quark to be promoted from the 0S shell to the P-shell - an energy of about 500 MeV. The spectrum of excited states can be fitted quite well with the same potentials and spatial dimensions of wavefunctions.

It was found, however, that the rms charge radius of the proton was too small in these calculations and the question arises as to how it can be increased. One possibility is that the proton wavefunction includes configurations in which the quarks are excited into higher shells. Parity must be conserved, so one quark may be excited from the 0S shell to the 1S shell (or higher), or two quarks could be excited into the 0P shell. Quite apart from a loss of potential binding energy, both these excitations require $2\hbar\omega$ of kinetic energy ($\approx$ 1000 MeV) and are thus strongly inhibited.

An alternative process is the spontaneous creation of pions (which have

free mass of only 140 MeV) in a P-state to conserve parity (as the pion has negative parity). These may be modelled as a $q\bar{q}$ pair and the most energetically favourable configuration is to have the $\bar{q}$ in the 0S shell along with 3 quarks and the fourth quark in the P-shell to give the correct parity. The energy required for this is low because of the strong attraction between the three quarks and the antiquark in the 0S shell compensates for the mass energy increase associated with the two additional particles. The numerical estimates are strongly interaction dependent, but Ayat [11] finds that, for the nucleon, $3q(q\bar{q})$ components contribute about 25% of the wavefunction while $3q$ excited states are an order of magnitude less. These $q\bar{q}$ components also improve the rms charge distribution. The length parameter can be too small to give the correct rms radius for a $(0S)^3$ configuration on its own, but a quark in the 0P shell increases the overall radius of the proton.

If antiquarks are pre-formed in the nucleon, it is a short step to speculate that they can couple with a quark to form a colourless pion-like object with a range unaffected by any colour confinement considerations. These objects cannot break free from the nucleon because this would violate conservation of energy, but they can exist for a time consistent with the Heisenberg uncertainty principle and this allows the possibility thet they are exchanged with another nucleon, giving rise to the possibility of an interaction mediated by colourless particles.

We will explore calculations of this kind in more detail at the end of chapter 2, after we have discussed the use of the shell model, which underpins our implementation of the quark potential model for the *NN* interaction.

Chapter 2

The Shell Model

2.1 Introduction

In general, it is not possible to obtain exact mathematical solutions to describe a many-body system of n interacting particles. In some cases, however, it is possible to approximate the interaction of one particle with all the others by an average central potential. The problem is then reduced to solving this single particle problem for each of the particles in the system. The description of a system of *interacting* particles in terms of independent particles moving in a central field is called a *shell model.*

In this chapter, we first review the development of the shell model for systems of electrons, nucleons and quarks. In nuclear physics, the shell model gives very good answers for some types of data, and poor answers in other cases. The results are briefly discussed as they give some indication of what may be expected in quark shell model calculations.

2.2 The Hartree Approximation

Perhaps the best example of a shell model can be seen in the electron system of an atom. The atomic electrons can be thought of as moving in the Coulomb potential due to the nuclear charge. The solution for a single electron moving in such a field will display a familiar hydrogen-like series of orbits. As more electrons are included in the system, occupying these orbits, the spacing and

order of these levels will change as the attractive field of the nucleus is modified by the mutual repulsion of the electrons. The Hartree theory [12, 13] provides a means of calculating the net central potential $V(r)$ experienced by each electron. A first approximation is made, namely

$$V(r) = \begin{cases} -\frac{Ze^2}{4\pi\epsilon_0 r} & r \to 0 \\ -\frac{e^2}{4\pi\epsilon_0 r} & r \to \infty \end{cases} \tag{2.2.1}$$

with some interpolation between the two values for intermediate r values. This is based on the assumption that an electron very near the nucleus feels the full Coulomb attraction of the charge $+Ze$, while very far from the nucleus it would feel a net charge of $+e$ due to the shielding effect of the charge $-(Z-1)e$ of the other electrons surrounding the nucleus. The Schrödinger equation is then solved for a typical electron, obtaining a set of single particle eigenstates. The ground state of the atom is then constructed by filling these with its electrons, giving the lowest energy value which satisfies the Pauli principle that no two electrons should occupy the same state. The charge distribution of the atom can then be evaluated as the sum of the individual $Z-1$ electron charge distributions [1] and that of the point nucleus, as seen by a typical electron.

Gauss's law is then used to calculate the electric field due to this distribution, allowing a new approximation to the original potential to be obtained. The above steps can then be repeated in an iterative process until a converged value of the net potential is obtained.

2.3 The Nuclear Shell Model

The Coulomb potential of the nucleus forms an obvious dominant potential for the multi-electron atom and it isn't too difficult to imagine that the system can be reasonably described by a net central potential obtained by modifying this. However, in the case of the nucleus, there is no equivalent potential and it is not immediately apparent that a central field will provide a good basis for a nuclear model. In fact, there was initially little hope that the application of a shell model to nuclei would be a success. The most popular nuclear models

[1] For each electron, this is just the product of the electric charge $-e$ and the probability density of the electron $\psi^*\psi$, (where ψ is the electron's eigenfunction).

were of the "liquid-drop" type and the idea of the nucleus as an unstructured drop of incompressible liquid was at odds with the central assumption of the shell model, namely that the nucleons mover more or less independently. The short range nature of the nuclear force made the idea of an average field acting on each nucleon seem highly unreasonable and it was thought that the motion of a nucleon would be heavily influenced by the effects of its nearest neighbours and essentially unaffected by nucleons more than a few fermis away.

Experimental results, though, did point towards some shell structure of the nucleus, particularly the discovery in the 1930's of so-called *magic numbers* of protons or neutrons

$$Z, N = 2, 8, 20, 28, 50, 82, 126 \tag{2.3.1}$$

which correspond to unusually stable nuclei. Nucleons with a magic number of protons and neutrons are exceptionally stable. This is directly analogous to the closed-shell effect displayed by the noble gases in atomic physics. It was also observed that the first proton or neutron beyond such a number was less tightly bound, again in direct analogy with atomic physics, where atoms with single electrons outside closed shells have the lowest ionization energy and are more chemically active.

Despite this evidence, no form of the central potential could be found which gave the correct ordering of energy levels to explain the magic numbers. A major advance was made in 1949 by Mayer, and independently by Jensen [14, 15] who introduced the concept of a nuclear spin-orbit interaction – much stronger than the equivalent interaction in atomic physics and opposite in sign. When combined with a harmonic oscillator or square well potential, the resultant splitting of the energy levels not only reproduces the magic numbers, but also gives a good description of many other nuclear properties. These single-particle energy levels have the characteristic of occurring in groups separated from each other by energy intervals which are large compared with the typical energy difference between the levels of each group. These groups are the *shells* which nuclear physicists refer to.

The single-particle potential experienced by a nucleon is thought of as an average potential resulting from all the two-body interactions between the nucleon and the others in the nucleus. Such a potential cannot completely

replace the physical interaction but can represent the main features.

2.4 Two-Body Interactions

Mayer and Jensen's shell-model identifies each nuclear level with a single configuration, corresponding to one nucleon in one single-particle orbit. It can be further improved by the addition of a two-body interaction as a perturbation to the single-particle Hamiltonian. This represents part of the total interaction experienced by the nucleons that is not described by the main central potential or the spin-orbit interaction. For this reason it is usually referred to as a *residual* interaction.

2.4.1 Configuration Mixing

Modern shell model calculations incorporate configurations that arise from protons and neutrons occupying several different partially filled orbits. The two-body interaction can change the orbits in which the particles move, giving rise to what is known as *configuration mixing*. The nuclear states are then no longer described by only one pure configuration. Instead, the nucleus has only a certain probability of being in a particular configuration for any given state. The most common approach is to apply the residual interaction to the valence nucleons [2] and treat those in filled shells as being part of an inert core. The implication is that the low energy properties of the nucleus are largely determined by the nucleons in the valence shell. In this model, the absolute binding energies are not usually of interest because they depend on features of the core which is not modelled in detail.

The inclusion of a residual interaction and configuration mixing produces a significant improvement in the results obtained using the shell model.

2.4.2 Form of the Two-Body Interaction

Two-body matrix elements for use in the shell model can be obtained in different ways. The most common approaches are

[2] The nucleons in levels beyond the last, completely filled shell

- To derive a nucleon-nucleon interaction from two-nucleon scattering data and the properties of the deuteron, or
- To fit the matrix elements to empirically known energy levels.

In the latter case, an explicit form of the interaction isn't needed and it has the additional advantage that the matrix elements are automatically optimised for the particular shell-model space that is in use. However, the number of parameters in such a process may be considerable – since there may be cases where the matrix elements are strongly dependent on the set of data being used, it is advisable to use many more data than the number of matrix elements being fitted. For example, Wildenthal has performed a fit to 447 energy levels of states in the sd-shell, which has 63 two-body matrix elements and obtains an rms error of only 200 keV per level. As might be expected, this method will generally give the best results in energy spectrum calculations. In addition, it is found, somewhat unexpectedly that the resulting wavefunctions also give good results for electromagnetic moments and transitions.

The other option is to start with a parametrized form of the free nucleon-nucleon interaction and attempt to fit this to match the phase shifts which are observed in nucleon scattering experiments and some of the observed properties of the deuteron, such as its binding energy. Unfortunately this leads to problems which arise because of the "hard core" nature of the nucleon-nucleon potential – *i.e.* it becomes highly repulsive at radial distances less than a certain core radius and tends to infinity as we approach the origin. This leads to divergent values of matrix elements when calculated with the unmodified single-particle wavefunctions. This problem can be avoided by carefully modifying the wavefunctions to make them vanish whenever the particles are within the hard core radius. Finally, the interaction must also take account, if possible, of the truncation of the model space (since higher shells will have some effect) and also of the presence of the "inert" core, which has an influence on the interaction between valence nucleons. Details of these complicated calculations can be found in [16].

It is because of these modifications that the interactions of this kind which are actually used in shell-model calculations are usually referred to as *effective* interactions.

2.4.3 Magnetic Moments

The nucleon has a magnetic dipole moment, $\boldsymbol{\mu}$, which is defined by the operator

$$\boldsymbol{\mu} = g_s \boldsymbol{s} + g_l \boldsymbol{l} \tag{2.4.1}$$

where g_s and g_l are the nuclear spin and orbital g-factors respectively[3]. This operator depends only on the way in which the spin and orbital angular momentum are combined to form the total angular momentum. When used to calculate the magnetic moments of various nuclei, it provides an excellent indication of the benefits of configuration mixing – the calculations in a full 1s0d-shell basis show a great improvement over the single-particle model, as shown in the top diagram of figure 2.1, which compares experimental results (y-axis) with those obtained from shell-model calculations (x-axis). The full-sd shell calculation results are shown on the right-hand side. This figure (bottom diagram) also shows that there is little difference between the results with the operator as used in its free-nucleon form (left-hand side) and an effective-operator version (right-hand side). In fact, the matrix elements calculated from the free-nucleon operator agree quite well with those predicted from experimental values of magnetic moments, though using the effective operator does give some improvement.

2.4.4 Electromagnetic Transition Rates

Electromagnetic transition operators depend on the spatial extent of the particle wavefunctions. For example, the E2 transition operator is proportional to

$$er^2 Y_{20}(\theta, \phi) \tag{2.4.2}$$

where e is the charge, and r, θ, ϕ are the spherical-polar coordinates of the particle. The results are thus strongly dependent on the radial wavefunctions and the use of harmonic-oscillator wavefunctions is inadequate – the results obtained are too small by a factor of about 4. The use of an oscillator basis is popular because it leads to an enormous simplification in the calculation

[3]These have different values for protons and neutrons, see for example [17].

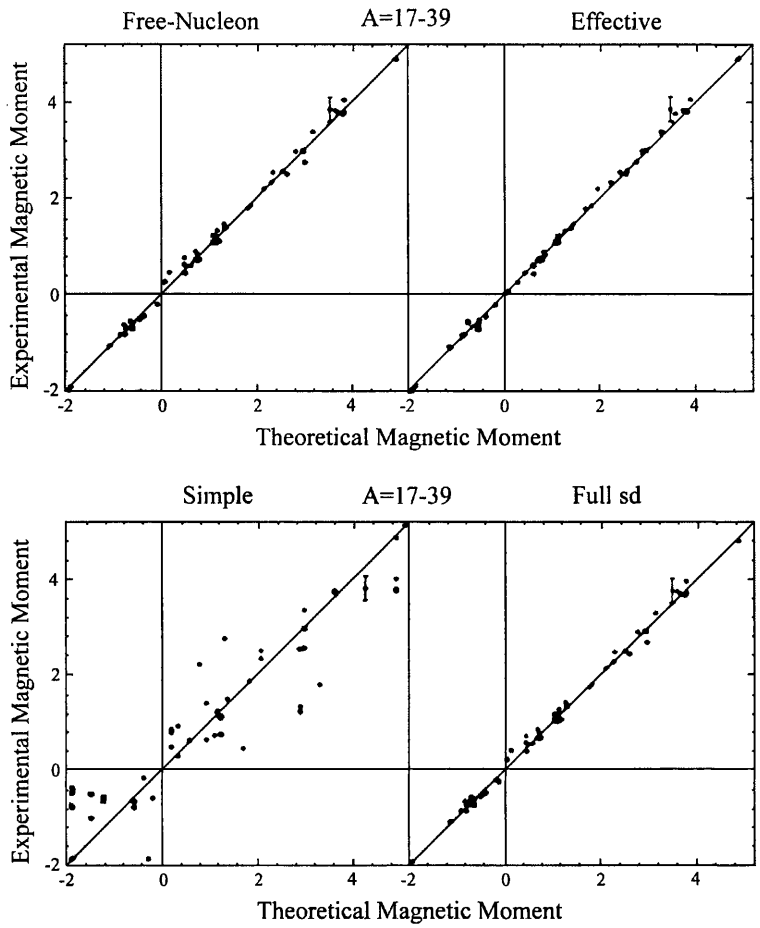

Figure 2.1: Shell-model calculations of nuclear magnetic moments, from reference [1]

of matrix elements, due to the fact that the system can be split into coordinates which describe the relative and centre-of-mass motion separately (see reference [18] and "the calculations in appendix A). The occasional excitation of particles into higher shells, or from the core have a large effect on $\langle r^2 \rangle$. This can be compensated for to some extent by the use of effective operators (most easily achieved by substituting effective charges for the free proton and neutron values) but the results are still not as good as those for nuclear magnetic moments, as can be seen from figure 2.2.

In a purely single particle model, two-body correlations are not well described, as we would expect. The infinite repulsive core in the *NN* potential

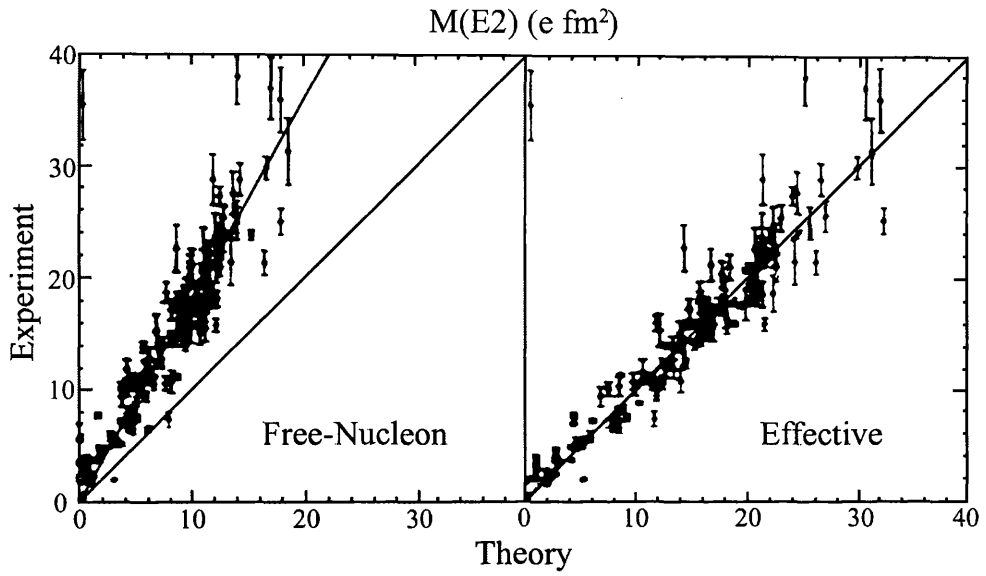

Figure 2.2: Shell-model calculations of E2 transition matrix elements, from reference [2]

does not allow two nucleons to occupy the same point in space, but a single-particle wavefunction does.

2.5 Quark Shell Models

In this work, we are applying the shell model framework to a completely different sphere of physics – the aim is to use it to perform calculations using the non-relativistic quark model. If we assume that a non-relativistic approach is justified in the first place (see section 1.5) then the shell model has some useful qualities which make it attractive for such calculations

- It provides a convenient means of handling a many-particle system – for a two-nucleon problem we have six quarks and when single-meson excitations are added, we can have up to 8 particles in the system.
- The Pauli Principle is automatically satisfied.

- The gluon-exchange potential between the quarks can be incorporated as a two-body shell model interaction.

In addition, quarks in the standard model are considered to be point-like particles, so the shell model postulate that they move essentially as free particles in an average potential is perhaps less open to criticism than in the nuclear case. In moving from electron to nucleon calculations, the shell model had to be extended to include isospin. In moving to quark model calculations, the model has to be extended to include colour, intrinsic parity and strangeness if necessary. An implementation of the model must also cope with basis states having different numbers of particles (if $q\bar{q}$ pairs are being included), which is not addressed in traditional nuclear physics.

The single-nucleon system is the most obvious starting point for calculations of this kind and, indeed, it has been shown that a reasonable description of the light baryon spectrum can be obtained by a shell-model NRQPM [4] treatment of a 3-quark system [11, 19, 20].

2.5.1 Nucleon-Nucleon Calculations

Attempts have also been made to describe two-nucleon system, using a six-quark system, also with some success. Investigations are often carried out in a similar fashion to that used in nucleon cluster models. An example would be the calculation of the interaction energy of two alpha particles [21]. The centres of the alpha particles are separated by a fixed distance, and a complete set of wavefunctions is constructed around each centre. Single particle orbits are then made from a linear combination of these wavefunctions [22]. The system is studied in a Born-Oppenheimer approximation which assumes that the relative motion between the two centres is slow compared with the internal motion of the quarks in each cluster [23].

Another technique which is frequently used in studying the NN interaction is the resonating group method (RGM) calculation. This is another method, originally developed by Wheeler [24] which was used in nuclear physics to directly calculate the phase shifts in scattering experiments. Good agreement

[4] Non-Relativistic Quark Potential Model (see section 1.5)

between the experimental and calculated results can be obtained for phenomena such as $\alpha\alpha$ scattering and the results are insensitive to the exact nature of the potential used to describe the interaction between the nucleons. It was then speculated that such techniques could also be used to describe nucleon-nucleon scattering, using two three-quark clusters and specifying supplying a potential for the quark-quark interaction. Full details of the complicated calculations involved can be found in [25].

Liberman [9], in 1977, was the first to publish results of a calculation of this kind. He found repulsion at short distances with cores of heights 350 MeV and 450 MeV for ^3S and ^1S interactions, respectively. He concluded that it was indeed possible to qualitatively reproduce the saturation property of nuclei, that is, the repulsive core at short range. However, his work was criticised for only taking into account the fully symmetric spatial representation with all six quarks placed in the 0S state and in 1981, Harvey [26] published results of his calculations which included configuration mixing – as well as the pure NN configuration, he included $N\Delta$, $\Delta\Delta$ and *hidden colour* states [5]. Unfortunately, this apparent improvement in the model space led to drastic reduction in the height of the repulsive core. In fact, with Harvey's full set of configurations included, he found a slight attraction in evidence at the origin. He concluded that, although configuration mixing seemed to be the physically correct way to proceed, it led to disappointing results.

The situation was remedied in further calculations and was blamed on Harvey's inappropriate choice of parameters for the model space he was using. In particular, the energy of the nucleon should be minimised with respect to the oscillator-length parameter used, the so-called "stability condition" [27]. The importance of this seems to depend on the nature of the basis used in the RGM calculations. Further calculations, including those done in Glasgow by Storm [28, 29] showed that the repulsive core could still be reproduced, although configuration mixing does reduce it substantially.

[5] States which are colourless overall, but consist of two coloured fragments

2.5.2 Meson Effects

One of the greatest drawbacks of quark cluster model calculations of this kind is that they are only capable of studying the very short range nature of the *NN* force, since the interaction is based purely on gluon exchange. It is already well established that the intermediate and long-range parts of the interaction are due mainly to the exchange of mesons so if we want to build a valid model of these regions, we cannot overlook the inclusion of meson effects. Many field theoretical studies of the *NN* interaction have been performed where the quark degrees of freedom are ignored and the nucleons and mesons are considered as elementary particles. In fact Yukawa's original theory of pion exchange mediating the nuclear force was inspired by the theory of quantum electrodynamics. This had shown that the long range of the Coulomb force is due to the fact that the photons which are the carriers of the force have zero rest mass. Yukawa adapted the theory to the case of the short-ranged nuclear force by assuming that the exchanged particle has a non-zero rest mass. The nuclear potential he derived can be written as

$$V(r) = -g^2 \frac{e^{-r/r'}}{r} \tag{2.5.1}$$

where

$$r' = \frac{\hbar}{\mu} c \tag{2.5.2}$$

The range, r', of the potential was approximately known from experiment ($\approx 1.5\,\text{fm}$), enabling the mass of the exchanged particle, μ to be estimated. The constant g^2 gives a measurement of the overall strength of the potential and can again be determined from experiment. This potential alone gives a good description of the long-range part of the nuclear force, since the pion is the lightest meson that can be exchanged. At medium range (2 - 4 fm), the situation becomes more complicated, as the effects of forces due to heavier mesons and of multi-pion exchange come into play. Early studies of the two-nucleon problem, for example [30], were usually content to treat this region phenomenologically to avoid over-complex calculations. Later, however, it was realised that multi-meson systems often displayed strong resonances and could satisfactorily be treated as a single boson. An example of this approach is the use of the so-called σ-meson which has been used to describe two-pion

exchange. These meson-field calculations have been continually refined over the years and a large body of work exists in this area. Detailed reviews are given in [31, 32].

For the short-range force between nucleons, meson-effects alone are no longer enough for a complete description. The structure of the nucleons themselves has to be included and this involves developing models based on the quark sub-structure of the nucleons, as discussed above. A lot of work has also been done by the likes of Faessler [33] and Oka and Yazaki [25], in combining the two approaches, leading to six-quark models supplemented by the addition of one and two-pion exchange potentials. These hybrid models have been able to give some remarkably good results, see for example [34, 35], but they are ultimately unsatisfactory in that they do not make a natural connection between the quark and mesonic degrees of freedom. We know that mesons are not fundamental particles, but are composed of quarks and antiquarks, so a full description must take this underlying structure into account.

An alternative approach has been adopted by Hecht and Fujiwara [36], who have applied the philosophy that a model in which baryons and mesons are described in terms of their common constituents is preferable to one in which quarks and meson fields are treated as separate entities. They have included mesonic degrees of freedom by incorporating quark-antiquark excitations directly into the model space. Potentials derived from the $q\bar{q}$ creation terms of the basic quark-gluon Lagrangian can then be used to couple the $6q$ configurations of the system to the new $6q(q\bar{q})$ components.

Hecht and Fujiwara's initial studies involved RGM calculations including single $q\bar{q}$ excitations in the model space. Their results showed a greatly reduced repulsive core from what would have been expected with an equivalent $6q$ model. There was also some evidence of medium range attraction though not enough to bind the deuteron. In later work, they extended these calculations to include two $q\bar{q}$ excitations, thus permitting the study of effects attributed to the σ and δ mesons of conventional meson field theory (see above).

2.5.3 Previous Work at Glasgow

The work in this thesis has its foundations in previous work carried out in the Glasgow group by Marion Storm [28, 29], Siraj Malik [37] and Leila Ayat [11, 38].

Storm's work was motivated by the studies of Harvey and his contemporaries, the aim being to investigate what contributions could be made to the field by applying the shell-model method originally developed at Glasgow for studying nuclei [39]. She performed calculations of the *NN* interaction with a six-quark system and showed that by choosing the oscillator parameter to minimise the nucleon mass, a repulsive core at the origin could be obtained, even with configuration mixing included.

As we have said above, the medium and long range part of the interaction cannot be described without including the mesonic degrees of freedom in the model. Ayat's work made use of the ideas of Hecht and Fujiwara, including $q\bar{q}$ excitations in a study of single baryon systems. She implemented a shell model which could deal with a variable number of particles and could handle antiquarks as well as quarks. She also extended the model to include strange quarks, which meant the model also had to be able to handle particles with different masses. She was able to perform calculations with up to two $q\bar{q}$ excitations before basis sizes became prohibitively large. She was able to obtain measurements of different configuration mixings; in the case of the nucleon, the model predicted a dominant $3q$ component, accounting for 73 % of the wavefunction, with $3q(q\bar{q})$ and $3q(q\bar{q})(q\bar{q})$ making up 22 % and 5 % respectively. It predicted hardly any strangeness in the nucleon wavefunction.

The aim of the present work is to combinine the models of Storm and Ayat, incorporating the quark-antiquark excitations into the two-centre calculations, and studying the effect they have on the *NN* interaction.

Mesons, explicitly included as $q\bar{q}$ pairs in the model give strongly interacting objects of low mass which can be exchanged over relatively long distances because they are colourless. They are the appropriate vehicles of the long range part of the interaction — other effects such as excitation of the core quarks to form Δ particles may be important but will not contribute much to the long range interaction, so it is still of value to perform calculations of

this kind without these other effects included in the model. In addition, it has already been shown in the work which preceded this study that the effects of expanding the shell-model space are substantially less than the effects of adding $q\bar{q}$ pairs. Ayat [11] found the nucleon mass to be 1249 MeV in the 0S space. Adding the 0P and (1S 0D) shells gave an energy of 1198 MeV and adding the (1P 0F) and (2S 1D 0G) shells reduced it further to 1129.7 MeV.

As we move towards applying the quark shell model to more complex systems, such as those which include meson configurations, the inadequacies of the approach are more likely to manifest themselves in the results. A model which gives reasonable results when used to study single baryons, or when treating the NN intraction as a six quark system may not be able to adequately describe the complexities of meson exchange between two baryons. Such effects are also more likely to depend on the spatial wavefunctions, so the inadequacy of the oscillator basis may become apparent, as in the case of the electromagnetic transition rates in the nuclear model (section 2.4.4).

Nevertheless, in the absence of more solid methods, it seems worthwhile pursuing such avenues of investigation as far as possible to find out just how much information on multiquark systems they are capable of providing.

Chapter 3

The Model and its Implementation

"My means are sane, my motive and my object mad".

Captain Ahab, Moby Dick.

3.1 Introduction

In this chapter we give a description of the main features of our model and the analytical and computational techniques required to carry out our calculations. More information as well as details of the program code used to implement the model can be found in the technical appendix C.

The basic process involves the construction of a Hamiltonian for our system of two baryons and its diagonalization to obtain the energy eigenvalues. This is carried out within the framework of the Glasgow Shell Model Code [39] which is characterised by the use of an "m-scheme" basis of Slater determinants for the model space and the use of the Lanczos Algorithm to find the eigenvalues.

Since we are dealing with two nucleons, the model uses a two centre approach. We use an oscillator basis for the spatial wavefunctions since this is physically reasonable for localised states and these wavefunctions greatly simplify the calculation of matrix elements for the problem, since it enables the use of relative and centre-of-mass coordinates (see appendix A). Three dimensional harmonic-oscillator wavefunctions are constructed around each centre

and the addition of spin, isospin, colour and intrinsic parity—to differentiate between quarks and antiquarks—gives a complete set of quantum numbers for the single particle basis states.

These single particle states are then combined to make up the many particle basis of Slater determinants. Each Slater determinant has the same z-component of spin, z-component of isospin, parity and has two each of red, green and blue quarks, and any number of $q\bar{q}$ pairs (which must be colour and anti-colour). Individual slater determinants are not eigenstates of $\boldsymbol{J}^2$, $\boldsymbol{T}^2$ and the colour Casimir operator $\boldsymbol{C}^2$.

The Hamiltonian can be written

$$H = \sum_i \frac{p_i^2}{2m_i} - \frac{P^2}{2M} + \sum_i m_i + \sum_{i<j} V(r_{ij}) \tag{3.1.1}$$

The first two terms are the kinetic energy and centre-of-mass kinetic energy operators. These are discussed in more detail below in section 3.13. Term three is just the sum of the masses of the particles in the state. In addition to these, there is a two-body interaction potential, derived from QCD, which describes the interaction of quarks by gluon exchange and the creation of $q\bar{q}$ pairs. The Hamiltonian conserves angular momentum, isospin, parity and colour, but not particle number, and the resulting eigenstates of H must be simultaneous eigenstates of $\boldsymbol{J}^2$, $\boldsymbol{T}^2$ and $\boldsymbol{C}^2$.

The integrals for the various spatial matrix elements are calculated by hand (see chapter 4 and appendix A) and are used by the program to build the matrix representation of the Hamiltonian in the basis of Slater determinants. The Lanczos algorithm is used to reduce this to tri-diagonal form and it is then diagonalized using standard methods.

Once the eigenstates have been found, the expectation values of $\boldsymbol{J}^2$, $\boldsymbol{T}^2$ and $\boldsymbol{C}^2$ are then evaluated to identify the states and also as a test for adequate convergence. Finally, projection operators are used to work out the number of particles and the quantum numbers for each cluster.

By performing the calculation for different cluster separations, the form of the interaction can be built up as a graph of the energy of the system against the two-centre separation.

3.2 The Single Particle Basis

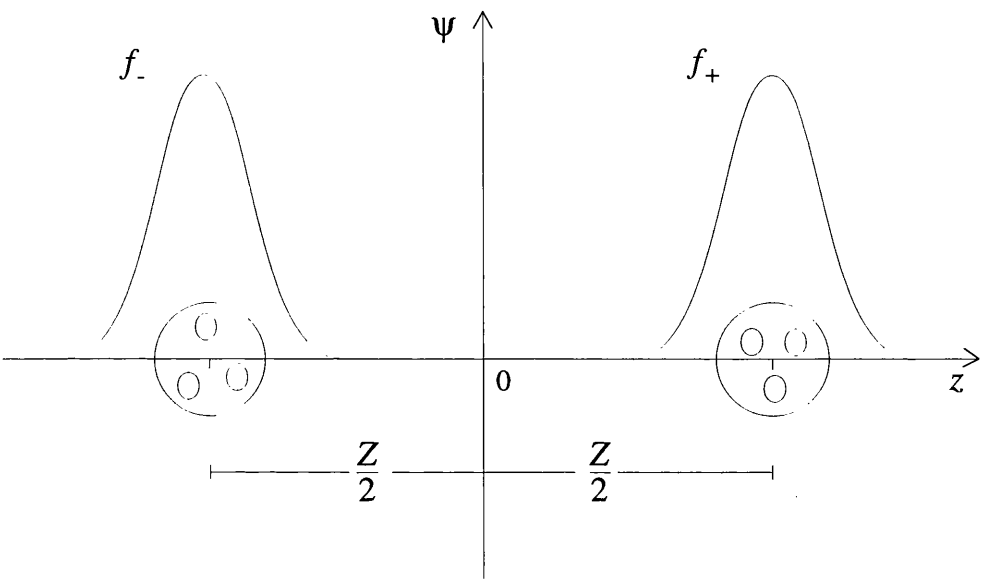

Figure 3.1: Illustration of the model

3.2.1 Spatial Wavefunctions

If we choose our co-ordinate system so that the two centres lie on the z-axis at co-ordinates $Z/2$ and $-Z/2$, then the normalised Gaussian wavefunctions for each of the clusters are

$$f_+ = \frac{1}{(\pi C^2)^{3/4}} \exp\left(-\frac{(\boldsymbol{x} - \frac{Z}{2}\boldsymbol{k})^2}{2C^2}\right) \tag{3.2.1}$$

$$f_- = \frac{1}{(\pi C^2)^{3/4}} \exp\left(-\frac{(\boldsymbol{x} + \frac{Z}{2}\boldsymbol{k})^2}{2C^2}\right) \tag{3.2.2}$$

"C" is the so called *oscillator length parameter* – one of the main parameters of the model. Here $\boldsymbol{x}$ is the position vector of the particle and "$\boldsymbol{k}$" is the unit vector in the direction of the z-axis. The situation is illustrated in figure 3.2.

As they stand, these are not orthogonal functions, and do not have definite parity, so we use a basis of the following normalised linear combinations of

them in our calculations

$$\phi_+ = N_+(f_+ + f_-) \tag{3.2.3}$$

$$\phi_- = N_-(f_+ - f_-) \tag{3.2.4}$$

with normalization factors $N_\pm$ given by

$${N_\pm}^2 = \frac{1}{2(1 \pm e^{-\frac{Z^2}{4C^2}})} \tag{3.2.5}$$

It should also be noted that ϕ_+ and ϕ_- have even and odd parity respectively. Transforming between the f and ϕ representations of the basis states is discussed in section 3.5.1.

3.2.2 Other Quantum Numbers

In addition to the spatial part, we require a string of other quantum numbers to represent a state completely: m_s, m_t are the z-projections of spin and isospin respectively, c—the colour quantum number—which has three degrees of freedom, labelled red, green and blue ($c \in \{r, g, b\}$ for quarks and $c \in \{\bar{r}, \bar{g}, \bar{b}\}$ for antiquarks) and a number for the intrinsic parity of the particle; the value 0 is assigned to a quark and 1 to an antiquark. If strange quarks are being included in the model space then the strangeness quantum number s is also required. Since we are only considering S-states there are no orbital angular momentum quantum numbers (l, m_l).
So the single particle basis states are

$$|\, q \,\rangle = |\, \phi_\pm m_s m_t c s 0 \,\rangle \tag{3.2.6}$$

for quarks, and

$$|\, \bar{q} \,\rangle = |\, \phi_\pm m_s \bar{m}_t \bar{c} \bar{s} 1 \,\rangle \tag{3.2.7}$$

for antiquarks. To avoid confusion, such single particle states will be referred to as *orbits*.

3.3 The Many Particle Basis

3.3.1 Number of Particles

In all but the most trivial cases (ie without $q\bar{q}$ creation), our calculations involve using basis states with different numbers of particles and the final eigenvector which we obtain to represent our system will be a mixture of these components. For example, the wavefunction of a nucleon-nucleon system will take the form of a linear combination of $6q$, $6q(q\bar{q})$, $6q(q\bar{q})^2, \ldots$ states:

$$\mid NN \rangle = \sum_{\alpha} C_{\alpha} \mid 6q \rangle + \sum_{\beta} C_{\beta} \mid 6q(q\bar{q}) \rangle + \sum_{\gamma} C_{\gamma} \mid 6q(q\bar{q})^2 \rangle + \ldots \qquad (3.3.1)$$

Here $\mid 6q \rangle, \mid 6q(q\bar{q}) \rangle, \ldots$ are normalised and rapidly become more complicated as the number of $q\bar{q}$ pairs increases. The coefficients C_{β} and C_{γ} provide a measurement of the importance in the wavefunction of the $q\bar{q}$ and $q^2\bar{q}^2$ components, respectively. In practice our ability to carry out calculations with two-meson excitations or more is limited, so the C_{γ}'s will usually be zero.

3.3.2 Slater Determinants

The standard approach in the Glasgow Shell Model Code is to use Slater determinants (SDs) to represent the many-particle basis states. In the usual second quantization notation (see section 3.6), a determinant consisting of n quarks and $\bar{n}$ antiquarks is represented by

$$b^{\dagger}_{\alpha_{n+\bar{n}}} b^{\dagger}_{\alpha_{n+\bar{n}-1}} \ldots a^{\dagger}_{\alpha_{n+1}} a^{\dagger}_{\alpha_n} a^{\dagger}_{\alpha_{n-1}} \ldots a^{\dagger}_{\alpha_1} \mid 0 \rangle \qquad (3.3.2)$$

Here $\mid 0 \rangle$ is the vacuum state and $a^{\dagger}$ and $b^{\dagger}$ are creation operators for quarks and antiquarks respectively and the αs represent a complete set of orbit quantum numbers. To satisfy the Pauli Principle, the wavefunctions used must be antisymmetric and SDs are guaranteed to be antisymmetric. They also have definite values of $M_s = m_{s_1} + m_{s_2} + \cdots + m_{s_{n+\bar{n}}}$ and $M_t = m_{t_1} + m_{t_2} + \cdots + m_{t_{n+\bar{n}}}$, but spin and isospin are not good quantum numbers.

When compiling our table of basis states, we need to take into account all acceptable configurations of the single particle orbits which can be linked by the transition potential. By "acceptable" we mean any combination of orbits

which have the specified quantum numbers (M_s, M_t and strangeness, S) and an appropriate number of quarks and antiquarks. These are chosen at the start of a calculation to match the system being studied.

As an example, consider how we construct the basis for a two-nucleon system. There are two possible spin values, S, for this system:

$$S = 0, \ with \ M_s = 0 \tag{3.3.3}$$

or

$$S = 1, \ with \ M_s = 1, 0, -1 \tag{3.3.4}$$

If we choose basis states with $M_s = 0$, we will have both $S = 0$ and $S = 1$ states represented in the calculation. This can lead to minor difficulties as they then have to be separated at the end of the calculation. This problem does not arise if we choose $M_s = 1$ (or -1), in which case the $S = 0$ state will not be obtained. Also, the number of basis states is smaller for $M_s = 1$ than for $M_s = 0$. If the states with $M_s = 0$ or -1 are required, they can be found from the one we have calculated by acting on it with lowering operators.

The selection of a value for M_t follows the same lines as choosing M_s. The number of particles in the basis states will depend on whether we are allowing meson creation or not. If, for example, we are considering allowing single $q\bar{q}$ pairs to be created, then we will have $n \in \{6, 7\}$ and $\bar{n} = n - 6$.

3.3.3 Coding of Slater Determinants

An SD for n particles is uniquely specified by the n labels of its occupied orbits and it can be stored by keeping a record of these occupation numbers. An obvious way of doing this is to map the orbits to the bits of a computer word and represent an occupied orbit by setting the corresponding bit of the word to 1 (see figure 3.2). This representation of the SD is efficient on storage space and is also very convenient when we come to apply creation and destruction operators to the SD.

0	0	0	0	1	0	0	...	0	0	0	1	0	1	0	1

Figure 3.2: Binary representation of a typical Slater Determinant.

Our calculations require up to 16 single particle orbits, without taking colour into account. So with colour included, a standard 32-bit computer word obviously isn't large enough to store the orbits we need for calculations in the full space. In practice three separate words are used—one to store the orbits for particles of each colour. Since the colour symmetry of the strong force is exact, any number which is an acceptable value for the red code will also be acceptable for the green and blue codes. So all three codes are obtained from the same set of numbers and we only have to generate this set for one colour, *e.g.* red

We are only interested in colourless states, so this places restrictions on the number of particles which can be in any single-coloured code — in basic six and three quark states, the particles must be evenly distributed between the three colour codes, *i.e.* there are two red, two green, two blue, or one red, one green, one blue respectively. Any $q\bar{q}$ pairs which are created must occur in a single colour code as $r\bar{r}$, $g\bar{g}$ or $b\bar{b}$. Furthermore, colour symmetry means that permutations of the colours r, g, b, should leave any colourless state unchanged. This is equivalent to saying that, if we expand a state $\mid \Psi \rangle$ in terms of our basis, the coefficient of a basis vector made up of three colour codes, R, G and B, $\mid RGB \rangle$, and the coefficients of the basis vectors corresponding to permutations of these three codes must be the same, *ie.* for

$$\begin{aligned} \mid \Psi \rangle = \ldots + C_1 \mid RGB \rangle + C_2 \mid RBG \rangle + C_3 \mid GRB \rangle + \\ C_4 \mid GBR \rangle + C_5 \mid BRG \rangle + C_6 \mid BGR \rangle + \ldots \end{aligned} \tag{3.3.5}$$

the values of $C_1, \ldots, C_6$ must be identical [1]. This being the case, it might seem reasonable to ask whether we actually need to consider all these basis states in our calculations. In fact, it is possible to reduce the basis to include just one of the states from the set of permutations when we are assembling the basis and compensate for the effects of the others later. To achieve this reduction, we impose the condition

$$R \leq G \leq B \tag{3.3.6}$$

[1] Here we have assumed R, G and B to be different so that all permutations of the colours do actually correspond to different basis states

when choosing the three codes making up one of the basis states to put in our list. States which do not satisfy this condition are referred to as *ghost* states and are not put in the list. Although these are not explicitly carried through the calculation, their presence is taken into account when expanding the eigenstates, when constructing the Hamiltonian and when calculating expectation values of J^2 and other operators. To do this, the only additional information we require is the multiplicity of each state which satisfies the colour constraint. This can be 6, 3, or 1, corresponding respectively to none, two or all of the colour codes being the same. This procedure reduces the size of the basis by a factor close to 6, making the calculation much easier. It also ensures that all the colourless eigenstates which would be found in the full space are retained, while most of the coloured ones are eliminated. Details of how the calculation is performed in the reduced basis are discussed in section 3.12.

3.3.4 Generating the Basis States

To summarise, each SD we generate must satisfy the following conditions:

- It must have the right quantum numbers: z-components of spin and isospin and correct parity (combination of intrinsic parities and the spatial parity of the orbital wavefunctions).
- It must obey our colour condition, equation 3.3.6.
- It must not violate the occupancy numbers for a particular shell. (In practice this is not important in our model since the particles are not restricted to particular shells in a full calculation)
- It must have an acceptable number of quarks and antiquarks.

The last constraint also applies to the individual colour codes; in practice we generate the set of all possible red codes and, since the sets of all possible green and blue codes are the same as this set, we combine the red codes in threes to give determinants which satisfy the above list of conditions.

The process is probably best illustrated with an example so we return to our calculation for a proton-neutron system, with $M_s = 1$ and allowing one meson to be created. Overall parity must be even. The basis states must have

at least six quarks and at most seven and the number of antiquarks can be zero or one. The number of red particles can be either two or four (since both particles in the created $q\bar{q}$ pair must have the same colour).

For the single particle orbits there are two possibilities for each of spin, isospin, spatial parity and intrinsic parity, giving 16 orbits in total. These are labelled such that the quarks occupy orbits 1-8 and the antiquarks lie in orbits 9–16. The red codes are produced by sequentially shifting first two, and then four bits through a word, rejecting any codes which have too many antiquarks and storing the rest and their associated quantum numbers. It is not possible to reject red codes on the basis that $M_s = 1$, for example, since M_s depends on the green and blue codes as well as the red one. Figure 3.3 shows the process pictorially.

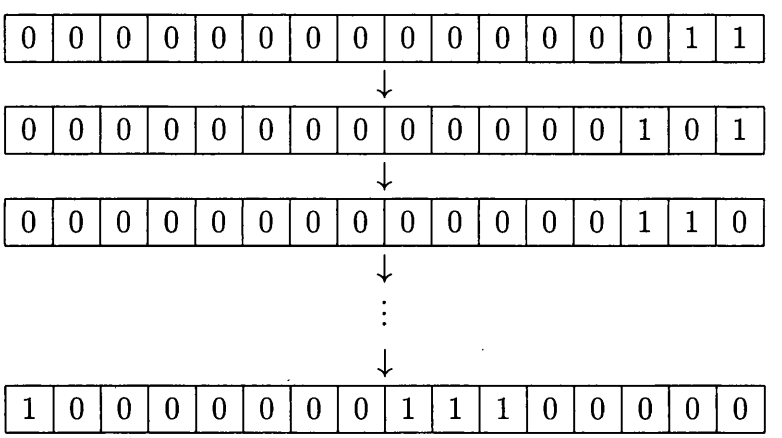

Figure 3.3: Generation of codes by bit shifting (two-particle case)

In this way we end up with a numerically ordered list of codes - there are ${}^8C_2 + {}^8C_3 \times 8 = 476$ of them. Any codes which contain an antiquark will automatically be higher up in the list than those which don't because of the way the orbits are numbered (the $\bar{q}$ orbits occupy the most-significant bits of the code).

We can then proceed to combine these single-colour codes in sets of three to make up the complete determinants; we pick a code for R, one for G and one for B. Since the codes are ordered numerically, it is very easy to implement the condition that $R \leq G \leq B$ and to maximise the efficiency of the program

code. At this stage, the other conditions on M_s etc. are tested, and if all the tests are passed, the state is included in the list of basis states.

In the particular example discussed above, there are 6426 basis states. The usefulness of the colour condition is immediately obvious when you consider that, without it, there are 37560 states!

3.4 Transitions Between Basis States

In this section we discuss the transitions which are possible as a result of the action of the Hamiltonian on the Slater determinants. As shown above in equation 3.1.1, the Hamiltonian contains several operators. These will be described in more detail when we come to talk about the actual evaluation of matrix elements but as far as we are concerned at the moment, they fall into two categories: those which conserve the number of particles in a state, and those which do not.

3.4.1 Number Conserving Operators

These involve the kinetic energy operator, the mass term and two-body transitions of the form $qq \rightarrow qq, q\bar{q} \rightarrow q\bar{q}$ and $\bar{q}\bar{q} \rightarrow \bar{q}\bar{q}$. The Pauli Principle must obviously be respected and, in addition, the transitions must also conserve spin, isospin, overall parity and both intrinsic and spatial parity separately. Colour must also be conserved so the colour of the created pair must be the same as that of the destroyed pair. However, this is not the full story. If a pair has the same colour and one particle is an antiquark, say $r\bar{r}$, then in addition to $r\bar{r} \rightarrow r\bar{r}$, we must also take into account the possibility of $r\bar{r} \rightarrow g\bar{g}$ and $r\bar{r} \rightarrow b\bar{b}$ which are allowed.

We proceed by constructing a table of sets of pairs of orbits, (α, β), with the same values of the above quantum numbers. We introduce the concept of *logical colour*: if we have $\langle c_3 c_4 \mid v_{12} \mid c_1 c_2 \rangle$ then we set $c_1 = 0$. The values of c_2, c_3, and c_4 can then be either 0 or 1, respectively, depending on whether they are the same or different colour as c_1. The permissible transitions for the $q\bar{q}$ case can then be expressed as $\langle 00 \mid v_{12} \mid 00 \rangle, \langle 01 \mid v_{12} \mid 01 \rangle$, and $\langle 11 \mid v_{12} \mid 00 \rangle$

The number of pairs from N single particle orbits will be $\frac{1}{2}N(N+1)$ of the same colour and N^2 different. We need some means of identifying any pair (i,j) so we can store information associated with it (set number, position of matrix elements etc.). To this end we set up the following mappings

$$(1,2) \rightarrow 1, (1,3) \rightarrow 2, \ldots, (2,3) \rightarrow N, \ldots, (N-1,N) \rightarrow N(N-1)/2 \quad (3.4.1)$$

if the colours are the same, and

$$(1,1) \rightarrow 1, (1,2) \rightarrow 2, \ldots, (2,1) \rightarrow N+1, \ldots, (N,N) \rightarrow N^2 \quad (3.4.2)$$

if the colours are different. This gives a unique number for each pair which can be used to address an array in which the set number is stored. In practice the quantity $\frac{1}{2}N(N+1)$ is added to the addresses for the different coloured pairs so that the same array can be used for both. The above mappings are equivalent to evaluating the following quantities

$$A_{\text{Same Colour}} = (2N-i)(i-1)/2 + j - i \quad (3.4.3)$$

$$A_{\text{Different Colour}} = N(N+1)/2 + N(i-1) + j \quad (3.4.4)$$

Records are also kept of the number of pairs in each set and the position in the list of pairs at which each set starts.

3.4.2 Number Non-Conserving Operator

This is the part of the interaction potential which permits meson creation and annihilation to take place, so transitions are of the form $q \rightarrow qq\bar{q}$, $qq\bar{q} \rightarrow q$. Since the numerical values of the matrix elements are the same for both types, we only calculate them for $q \rightarrow qq\bar{q}$. If two meson states are included in the model space we would also have to consider transitions like $\bar{q} \rightarrow \bar{q}q\bar{q}$, but this is not possible at the moment because the basis sizes are too large.

In this case we have to make up sets of *triples* associated with the destruction of a particle in each of our N orbits. Spin and isospin must be conserved, as above. However, with the creation of the extra antiquark, it is necessary that the spatial and intrinsic parities of the created triple are opposite to those of the destroyed particle in order to conserve parity overall. Colour transitions will be of the form $0 \rightarrow 000$ and $0 \rightarrow 011$, in the logical colour notation described in the previous section.

3.5 Calculating Matrix Elements

The next chapter describes the calculation of the matrix elements for the various operators used in the model, so this section is just intended to give an outline of the procedure to clarify what follows.

As things stand, we have our sets of pairs and we require some means of calculating the matrix elements for the allowed transitions and storing them.

For each set, we locate the first pair in the list which belongs to it and step through each pair in turn, working out the two body matrix elements between this and all the subsequent pairs in the set and storing them sequentially. For each of these pairs, a pointer to the position of the first of its matrix elements is recorded. This makes it easy to recover the sequence when we actually come to use the matrix elements.

3.5.1 $f \rightarrow \phi$ Transformation

When it comes to calculating our matrix elements, the integrals are performed in the f_μ basis, as defined in section 3.2. However, the basis states, ϕ_μ, are used in the programs so we need to be able to switch easily between expressing one and two particle states in both the f and ϕ representations. This can be done using the following transformation matrix

$$T = \begin{bmatrix} N_+ & N_+ \\ N_- & -N_- \end{bmatrix} \tag{3.5.1}$$

So that

$$\begin{bmatrix} \phi_+ \\ \phi_- \end{bmatrix} = T \begin{bmatrix} f_+ \\ f_- \end{bmatrix} \tag{3.5.2}$$

If Ω is a one body operator and Φ is the matrix $\langle \phi_\mu \mid \Omega \mid \phi_\nu \rangle$, and F is the matrix $\langle f_\mu \mid \Omega \mid f_\nu \rangle$, then the one body matrix elements transform as

$$\Phi = TFT^T \tag{3.5.3}$$

where T^T is the transpose of T.

The situation with two body matrix elements is a bit more complicated because four single-particle states have to be transformed, but the method is obvious.

3.5.2 Antisymmetrisation of Matrix Elements

The use of antisymmetrised wave functions in the calculation means that the two-body matrix elements must be properly antisymmetrised [40]. The antisymmetric form of a two body wavefunction $\mid \alpha_1, \alpha_2 \rangle$, is given by

$$\mid \alpha_1, \alpha_2 \rangle_{\text{Antisymmetric}} = \frac{1}{\sqrt{2}} [\mid \alpha_1 \alpha_2 \rangle - \mid \alpha_2 \alpha_1 \rangle] \tag{3.5.4}$$

with the α's representing the set of single particle quantum numbers. To work out our matrix elements we have to sandwich our two-body operators between two such wavefunctions. So the antisymmetrised form of a matrix element $\langle \alpha_3 \alpha_4 \mid V \mid \alpha_1 \alpha_2 \rangle$ would be given by

$$\begin{aligned} \langle \alpha_3 \alpha_4 \mid V \mid \alpha_1 \alpha_2 \rangle_{\text{Antisymmetric}} = \frac{1}{2} [&\langle \alpha_3 \alpha_4 \mid V \mid \alpha_1 \alpha_2 \rangle \\ &- \langle \alpha_4 \alpha_3 \mid V \mid \alpha_1 \alpha_2 \rangle - \langle \alpha_3 \alpha_4 \mid V \mid \alpha_2 \alpha_1 \rangle + \langle \alpha_4 \alpha_3 \mid V \mid \alpha_2 \alpha_1 \rangle] \end{aligned} \tag{3.5.5}$$

If we take advantage of the fact that $\langle \alpha_k \alpha_l \mid V \mid \alpha_i \alpha_j \rangle = \langle \alpha_l \alpha_k \mid V \mid \alpha_j \alpha_i \rangle$, then the outcome is that a matrix element can be expressed as the difference between a *direct* and an *exchange* term

$$\langle \alpha_3 \alpha_4 \mid V \mid \alpha_1 \alpha_2 \rangle_{\text{Antisymmetric}} = \underbrace{\langle \alpha_3 \alpha_4 \mid V \mid \alpha_1 \alpha_2 \rangle}_{\text{Direct}} - \underbrace{\langle \alpha_3 \alpha_4 \mid V \mid \alpha_2 \alpha_1 \rangle}_{\text{Exchange}} \tag{3.5.6}$$

The presence of the exchange term takes care of the antisymmetrisation; if $\mid \alpha_1 \rangle = \mid \alpha_2 \rangle$, the matrix element is zero.

3.6 Second Quantization

The technique of second quantization provides a useful tool for use in many-body calculations. In the basic formalism, a many particle state is manipulated by means of *creation* and *annihilation* operators which create and destroy individual particles within the state.

A creation operator $a^\dagger_\alpha$ acting on a state $\mid \phi \rangle$, produces an additional particle in the orbit α within that state. In particular, if $\mid \phi \rangle$ is the vacuum state, then $a^\dagger_\alpha \mid \phi \rangle$ represents the single-particle state, $\mid \alpha \rangle$. If $\mid \phi \rangle$ is an SD

containing N particles, then $a_\alpha^\dagger \mid \phi\rangle$ represents a new $N+1$ particle state, with the orbit α added to the orbits which were occupied in $\mid \phi\rangle$. The conjugate operator a_α, called the annihilation operator for the state α, removes a particle from the orbit α in the state to which it is applied. So if $\mid \phi\rangle$ is an N particle state as before, then $a_\alpha \mid \phi\rangle$ represents the $N-1$ particle state obtained by eliminating the orbit α from the state $\mid \phi\rangle$.

The following special cases should be noted — if the orbit α is occupied, then to satisfy the Pauli Exclusion Principle[2] we must have

$$a_\alpha^\dagger \mid \phi\rangle = 0 \tag{3.6.1}$$

because we can't produce another particle in a state which is already occupied. Similarly, if the orbit α is unoccupied, we cannot destroy a particle in it, so

$$a_\alpha \mid \phi\rangle = 0 \tag{3.6.2}$$

It can be shown from these definitions, and the requirement that the wavefunctions be antisymmetric, that the creation and annihilation operators satisfy the following anti-commutation[3] relations [41]

$$\{a_\alpha, a_\beta\} = 0 \tag{3.6.3}$$

$$\left\{a_\alpha^\dagger, a_\beta^\dagger\right\} = 0 \tag{3.6.4}$$

$$\left\{a_\alpha, a_\beta^\dagger\right\} = \delta_{\alpha\beta} \tag{3.6.5}$$

It follows that whenever we exchange two destruction or two creation operators we get a minus sign (which arises from the complete antisymmetry of our wavefunctions and corresponds to exchanging columns in a Slater determinant). Because of this, there is a phase associated with the order in which we apply our operators. We must always arrange them to give some standard order and keep a note of the sign changes this involves.

In our case, we take the standard order to be that in which both creation and annihilation operators having ascending orbit number from left to right. This is different from the order given in most text books where the annihilation operators are usually sorted into descending order. This choice lets us calculate the phase more easily as we shall see below.

[2]Second quantization techniques can also be applied to bosons but we are obviously talking about fermions here.

[3]The anti-commutator of a and b, $\{a, b\}$, is defined by $\{a, b\} = ab + ba$.

0	0	0	0	0	0	0	0	1	0	1	1	0	1	1	1

Figure 3.4: Slater determinant with orbits 1,2,3,5,6 and 8 occupied.

As an example, suppose we have a six particle SD, $\mid \phi \rangle$ as shown in figure 3.4. This has particles in orbits 1,2,3,5,6 and 8. We can write it as

$$\mid \phi \rangle = a_1^\dagger a_2^\dagger a_3^\dagger a_5^\dagger a_6^\dagger a_8^\dagger \mid 0 \rangle \tag{3.6.6}$$

where $\mid 0 \rangle$ is the vacuum state. If, for example, we than apply a two-body operator with a term $a_5^\dagger a_7^\dagger a_2 a_5$, then we have

$$\begin{aligned} a_5^\dagger a_7^\dagger a_2 a_5 \mid \phi \rangle &= a_5^\dagger a_7^\dagger a_2 a_5 a_1^\dagger a_2^\dagger a_3^\dagger a_5^\dagger a_6^\dagger a_8^\dagger \mid 0 \rangle \\ &= +a_5^\dagger a_7^\dagger a_1^\dagger a_3^\dagger a_6^\dagger a_8^\dagger \mid 0 \rangle \\ &= -a_1^\dagger a_3^\dagger a_5^\dagger a_6^\dagger a_7^\dagger a_8^\dagger \mid 0 \rangle \end{aligned} \tag{3.6.7}$$

So the overall phase is negative. Determining these phase factors is quite a tricky business and we use a special representation to calculate it. This is explained in section 3.7.

In second quantization notation, a one-body operator is

$$F = \sum_{\alpha\beta} \langle \alpha \mid f \mid \beta \rangle a_\alpha^\dagger a_\beta \tag{3.6.8}$$

and a two-body operator is written

$$G = \sum_{\alpha\beta\gamma\delta} \langle \alpha\beta \mid g \mid \gamma\delta \rangle a_\alpha^\dagger a_\beta^\dagger a_\delta a_\gamma \tag{3.6.9}$$

In addition to these two standard forms, we also have our $q \rightarrow qq\bar{q}$ interaction which is of the form

$$H = \sum_{\alpha\beta\gamma\delta} \langle \alpha\beta \mid h \mid \gamma\bar{\delta} \rangle a_\alpha^\dagger a_\beta^\dagger a_\delta^\dagger a_\gamma \tag{3.6.10}$$

This is discussed in more detail in section 4.3.

3.7 Parity Representation

In constructing the many-body matrix elements from the two-body ones, there is a phase factor associated with the ordering of the creation and destruction

operators acting on the state, as explained in section 3.6. The sequence in which they are applied must be compared with the "standard" order. Swapping the operators around to match this order introduces the phase and the use of the so called *parity representation*, instead of the usual occupancy representation was introduced in the Glasgow Shell Model code to allow the phase to be calculated as efficiently as possible [42, 43]. This is of primary importance when the Hamiltonian is too large to be stored and has to be re-calculated bit-by-bit for each iteration of the Lanczos algorithm. In these circumstances, any additional overheads in the code can substantially increase the time required for the calculation.

In our case, the matrices we are dealing with are not overly large and only need to be calculated once, so the speed of the phase calculations is not of paramount importance. Nevertheless, we still use an implementation of the parity representation since it provides a suitably elegant method of finding the phase and allows for the possibility of future expansions of the code when the Hamiltonian may have to be rebuilt for each iteration. There are several ways of implementing the parity representation and our approach is not the same as that given in [42].

To form the parity representation of a SD, we start by setting the least significant bit to zero. We then work our way up through the corresponding bits of the occupancy representation. Every time a filled orbit is encountered, the state of the bits in the parity representation is toggled immediately after it. So the bits will be low until the first occupied orbit is found, they then change to high immediately after it, back to low after the next occupied orbit and so on.

When an operator is applied in second quantization notation, it must be moved to the correct position in the sequence of operators and each time it is moved past another operator, the phase toggles from + to -. The toggling of the bits in the parity representation corresponds to this swapping of operators to achieve the standard order. We can work out the phase of an operator by examining the corresponding bit in the parity representation. If the bit is low, the phase is even, if it is high, the phase is odd.

In the case of a two-body operator, the phase due to both destroying and creating two particles can be worked out by comparing the two bits corre-

sponding to the two destroyed or created orbits. If they are the same, the phase is even; if they are different, it is odd.

As an illustration, we use the example from section 3.6 again, where the state $\mid \phi \rangle$ has orbits 1,2,3,5,6 and 8 occupied — refer back to figure 3.4. We first construct the parity representation of this SD. The bit corresponding to orbit 1 is set to zero. Orbit 1 is occupied so bit 2 is set high. Orbit 2 is also occupied, so the parity rep. toggles back to zero at bit 3. Orbit 3 is occupied and bit 4 toggles back to high again. The next filled orbit is orbit 5, so the bits remain high until bit 6 which is low. Bit 7 goes back to high, because orbit 6 is filled and we remain high until immediately after the last occupied orbit, number 8. Thereafter the bits return to low and remain so.

The resulting parity representation is shown at the top of figure 3.5. The subsequent diagrams show its alteration and the phase calculations, as the operators from the example are applied.

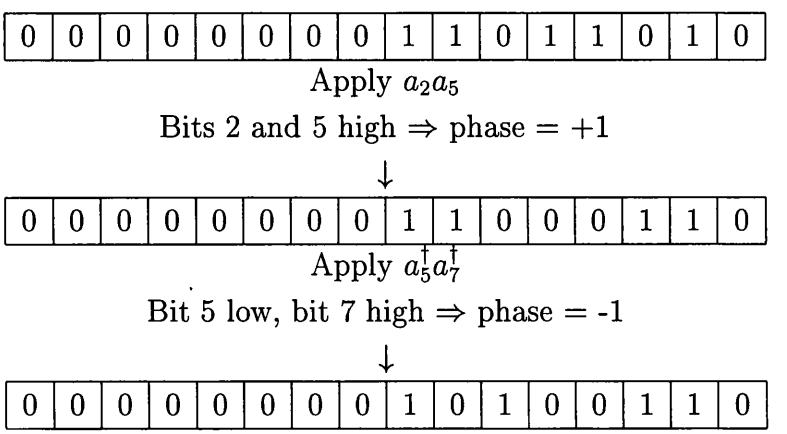

Figure 3.5: Operators and the Parity Representation

For an SD with one single occupied orbit a, say, the parity representation is given by $(1 \ll a) - 1$. For many particle SD's, it is easily formed by working out the parity representations of the constituent orbits as single-particle states and constructing the **XOR** of these [4]. The application of annihilation operators is performed in the same way—using the parity representation of the destroyed orbit.

[4] bitwise exclusive **OR** operator

In a language like 'C' which provides a comprehensive set of bitwise operators [44], these manipulations of the parity representation can easily be coded very efficiently.

3.8 Building the Hamiltonian

A matrix element of the Hamiltonian, $\langle j \mid H \mid i \rangle$, say, is found by adding up all the elements from the transitions which correspond to changing state $\mid i \rangle$ into state $\mid j \rangle$. By doing this for all i and j in the basis we can assemble the full matrix.

With this in mind, we proceed as follows:

1. Select the SD for state i.
2. Destroy a pair of particles in this SD.
3. Create all the pairs from the same set in turn.
4. Check that the new state formed by creation of each of these pairs is valid (*i.e.* that none of the orbits we are trying to create are already occupied).
5. Locate the new state in the basis table.
6. Store the number of the state and a pointer to the matrix element associated with the transition.

Here the two-body case has been used as an example. The process is very similar for the $1 \rightarrow 3$ case. Creation and destruction of particles is easily performed by the use of bitwise operations which are also very economical on computer time. To destroy a pair of particles, we shift bits to the appropriate positions in a word and form the exclusive-OR of this and the SD. A similar word can be set up for the pair we want to create and step 4 can be performed by taking the bitwise AND of this and the remnant of the SD. If this is zero, then neither of the orbits is already occupied. Finally, the pair can be inserted in the SD by an OR operation. Things are obviously a bit more complicated

than this because we actually have three separate SDs—one for each colour—but the general principle is the same.

Step 5 is best done by using a binary search. This is easy because the basis is stored in numerical order. It is possible that a ghost state is formed, so the three codes of the newly created SD may not immediately satisfy the colour condition given in equation 3.3.6 and they have to be ordered correctly before performing the search.

3.9 The Hamiltonian Matrix

It has already been pointed out that the number of basis states in our calculations can run into thousands (or tens of thousands!), especially when we want to use a model space with meson excitations. At first sight, this might appear to make the Hamiltonian too large to handle since it will have hundreds of thousands of elements. However, we are saved by the fact that most of the matrix elements between basis states are in fact zero. In other words the matrix is very sparse. This is illustrated by the plot in figure 3.6 which shows the intriguing structure of the matrix—in this case with 648 basis states. Non-zero elements are represented by points and the white space represents the zeroes.

This means that the number of elements we have to store is greatly reduced. In the present implementation, the matrix is stored row by row. Each non zero element is put in a file along with its position in the row.

3.9.1 Form of the Matrix

The matrix obtained above is actually complex Hermitian due to the nature of the interaction potential for $q \rightarrow qq\bar{q}$ (See section 4.3, below). If we ordered the basis, sorting it by the number of particles in each state, the matrix would take on the form

$$H = \begin{bmatrix} M_1 & iM_2 \\ -iM_2^T & M_3 \end{bmatrix} \tag{3.9.1}$$

Where M_1, M_2, M_3 are real matrices and M_1 and M_3 are symmetric. The basis states from 1 to $\dim(M_1)$ are $6q$ configurations and those from $\dim(M_1)+1$

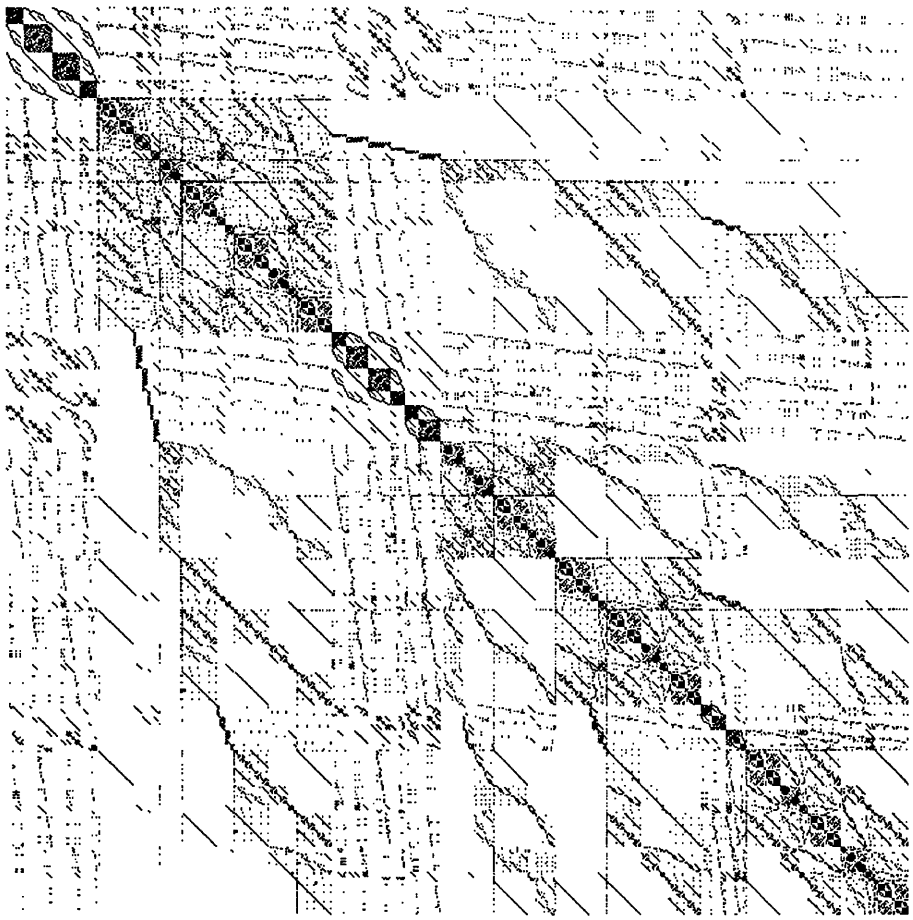

Figure 3.6: Sparsity plot of a typical Hamiltonian Matrix

to $\dim(M_1) + \dim(M_3)$ are $6q(q\bar{q})$. We can write H as

$$H = \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix} \begin{bmatrix} M_1 & M_2 \\ M_2^T & M_3 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \tag{3.9.2}$$

which we can write as

$$H = P^{-1}AP \tag{3.9.3}$$

where

$$A = \begin{bmatrix} M_1 & M_2 \\ M_2^T & M_3 \end{bmatrix} \tag{3.9.4}$$

and

$$P = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \tag{3.9.5}$$

or in other words, H is similar to A [45] (Clearly $P^{-1}P = I$, the identity matrix). Similar matrices have the same eigenvalues [5], so we do not actually have to worry about the complex nature of H. Instead we just deal with the matrix A, which we refer to as H from now on.

The coefficients of the SDs in an eigenvector of A are purely real numbers. Those in the corresponding eigenvector of H have the same numerical values, but some are real and some become pure imaginary, depending only on the number of antiquarks in the SD. If $\boldsymbol{u}$ and $\boldsymbol{v}$ are vectors associated with the matrix A the $P\boldsymbol{u}$ and $P\boldsymbol{v}$ are the corresponding vectors associated with H. Since $(P\boldsymbol{v})^\dagger(P\boldsymbol{u}) = \boldsymbol{v}^T P^{-1} P\boldsymbol{u} = \boldsymbol{v}^T\boldsymbol{u}$, scalar products are the same in both cases.

Since we are only interested in matrix elements, this irritation can be ignored entirely when performing the calculations.

3.10 The Lanczos Algorithm

Once we have the Hamiltonian safely under wraps, we are ready to find the eigenstates of the system using the iterative Lanczos Algorithm [46, 47]. Lanczos diagonalization of the Hamiltonian is one of the central features of the Glasgow Shell Model Code, pioneered by Whitehead et. al [39] for use in full-basis calculations for sd-shell nuclei and now used in all other large scale shell model codes. Its use is almost essential if we are realistically to deal with the very large matrices which calculations of this kind produce.

The main attraction of using the Lanczos algorithm is that it gives extremely rapid convergence of the lowest eigenvalues (*ie.* the ones we want) with a small number of iterations and it is unnecessary to diagonalize the matrix completely. If we choose a normalised starting vector $\boldsymbol{v}_1$ in the model space (of dimension n), then the algorithm proceeds as follows, with $i = 1, 2, \ldots, n$

[5] We have

$$H - \lambda I = P^{-1}AP - \lambda I = P^{-1}AP - \lambda P^{-1}IP = P^{-1}(A - \lambda I)P \tag{3.9.6}$$

So

$$|H - \lambda I| = |P^{-1}||A - \lambda I||P| = |A - \lambda I||P^{-1}P| = |A - \lambda I| \tag{3.9.7}$$

i.e. H and A have the same eigenvalues.

and $\beta_0 = 0$:

$$\left.\begin{aligned} \alpha_i &= \boldsymbol{v}_i \cdot \boldsymbol{H}\boldsymbol{v}_i \\ \boldsymbol{u}_{i+1} &= \boldsymbol{H}\boldsymbol{v}_i - \alpha_i \boldsymbol{v}_i - \beta_{i-1}\boldsymbol{v}_{i-1} \\ \beta_i &= \sqrt{\boldsymbol{u}_{i+1} \cdot \boldsymbol{u}_{i+1}} \\ \boldsymbol{v}_{i+1} &= \beta_i^{-1}\boldsymbol{u}_{i+1} \end{aligned}\right\} \tag{3.10.1}$$

This generates a set $\{\boldsymbol{v}_i\}$ of orthonormal basis vectors, called Lanczos Vectors, by repeatedly operating with the matrix $\boldsymbol{H}$, and projecting out the components orthogonal to the previous two vectors. For this it is useful to note that, because $\boldsymbol{H}$ is Hermitian,

$$\begin{aligned} \beta_i &= \boldsymbol{v}_{i+1} \cdot \boldsymbol{H}\boldsymbol{v}_i \\ &= \boldsymbol{v}_i \cdot \boldsymbol{H}\boldsymbol{v}_{i+1} \end{aligned} \tag{3.10.2}$$

so the component of $\boldsymbol{H}\boldsymbol{v}_{i+1}$ in the direction $\boldsymbol{v}_i$ has magnitude β_i. It is only necessary to remove components in the directions of the two preceding vectors because all others are automatically zero, in the absence of numerical rounding errors. This is immediately apparent, because

$$\begin{aligned} \boldsymbol{v}_{i-2} \cdot \boldsymbol{H}\boldsymbol{v}_i &= \boldsymbol{v}_i \cdot \boldsymbol{H}\boldsymbol{v}_{i-2} \\ &= \boldsymbol{v}_i \cdot (\beta_i \boldsymbol{v}_{i-3} + \alpha_{i-2}\boldsymbol{v}_{i-2} + \beta_{i-2}\boldsymbol{v}_{i-1}) \end{aligned} \tag{3.10.3}$$

and this evaluates to zero since $\boldsymbol{v}_i$ is orthogonal to $\boldsymbol{v}_{i-1}, \boldsymbol{v}_{i-2}$ and $\boldsymbol{v}_{i-3}$. This also means that the matrix elements of $\boldsymbol{H}$ in the new basis are given by

$$\boldsymbol{H}'_{ij} = \boldsymbol{v}_i \cdot \boldsymbol{H}\boldsymbol{v}_j = \beta_{i-1}\delta_{ij-1} + \alpha_i \delta_{ij} + \beta_i \delta_{ij+1} \tag{3.10.4}$$

So the matrix in this basis takes on a tri-diagonal form:

$$\boldsymbol{H}' = \begin{bmatrix} \alpha_1 & \beta_1 & & & & \\ \beta_1 & \alpha_2 & \beta_2 & & & \\ & \beta_2 & \alpha_3 & \beta_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & \beta_{n-2} & \alpha_{n-1} & \beta_{n-1} \\ & & & & \beta_{n-1} & \alpha_n \end{bmatrix} \tag{3.10.5}$$

Since the Lanczos vectors form a complete, orthonormal basis spanning the space, $\boldsymbol{H}'$ has the same eigenvalues as $\boldsymbol{H}$. It is also the case that the eigenvalues of the leading submatrices of $\boldsymbol{H}'$ converge very rapidly to the extremal

eigenvalues of the full matrix as its dimension increases with successive iterations. In practice, this means that we rarely need more than 100 iterations to get the answers we need.

3.10.1 Re-Orthogonalisation

In the world of real calculations, of course, things are rarely so simple and the assumption that the Lanczos vectors are orthogonal is likely to prove false because of rounding errors. The effects of this can be minimised by re-orthogonalising each new vector to all the previous ones. In addition, the orthogonalisation routine and accumulation of scalar products are always carried out with double precision numbers.

3.10.2 Choice of Starting Vector

So far, nothing has been said about the choice of the Lanczos vector we begin with, $\boldsymbol{v}_1$. It is possible to greatly reduce the number of iterations necessary by choosing a vector which contains a substantial component in the direction of an eigenstate. At first, this may seem like a rather useless piece of information since the eigenstates are what we are trying to find! However, we do know that the $6q$ part of an eigenfunction will be much stronger than the part with $q\bar{q}$ excited components, so if we take a note of the $6q$ states in the basis and set only these elements in $\boldsymbol{v}_1$ this should go some way to improving the rate of convergence.

An alternative method of fine-tuning the algorithm is to use an eigenvector from a previous calculation for the same system at a different separation. It is a reasonable assumption that a small change in the separation should not cause too drastic a change in the eigenvector. This approach does in fact make the calculation much more efficient than using a random vector. We simply store the ground state for each calculation and use it as our starting vector for the subsequent one. This is especially effective when we are only interested in finding the ground state. It might be feared that this could result in slow convergence for other low lying orthogonal states, but this has not been a problem.

There may also be some concern that it is possible to accidentally start with an exact eigenvector of our system because in this case the algorithm will break down at the first step. In practice, though, this is highly unlikely because of rounding errors. Even if it did happen, we could just alter the vector slightly and repeat the calculation.

3.11 Eigenstates of $\boldsymbol{H}$

We find the eigenvalues of $\boldsymbol{H}'$ (and $\boldsymbol{H}$) by first obtaining an interval in which the desired eigenvalues lie, and then pinning them down to the required accuracy by using bisection combined with the Sturm sequence property described below [48, 49, 50]. This approach is efficient enough when we only want to find a few eigenstates.

3.11.1 Sturmian Sequence Property

With $\boldsymbol{H}'$ defined as in equation 3.10.5, then the maximum value $N_{\max}$ in the sequence of rownorms, of $\boldsymbol{H}'$,

$$\alpha_1 + \beta_1, \alpha_2 + \beta_2 + \beta_1, \alpha_3 + \beta_3 + \beta_2, \ldots, \alpha_n + \beta_{n-1} \tag{3.11.1}$$

provides an upper and lower bound for our eigenvalue search [51] – they lie in the interval $[-N_{\max}, N_{\max}]$. The characteristic polynomial $p_r(\lambda)$ of the rth leading sub-matrix of $\boldsymbol{H}'$ can be found from the recursion relation

$$p_r(\lambda) = (\alpha_r - \lambda)p_{r-1}(\lambda) - \beta_r^2 p_{r-2}(\lambda) \tag{3.11.2}$$

where $p_0(\lambda) = 1$ and $p_1(\lambda) = \alpha_1 - \lambda$. For any value of λ, the number of sign changes in the sequence

$$p_0(\lambda), p_1(\lambda), \ldots, p_n(\dot{\lambda}) \tag{3.11.3}$$

is equal to the number of eigenvalues of $\boldsymbol{H}'$ which are less than λ. This property can be combined with a binary search method to pin down any particular eigenvalue of the matrix.

3.11.2 Eigenvectors

The eigenvector $\boldsymbol{x}$ corresponding to the eigenvalue λ is found by solving

$$(\boldsymbol{H}' - \lambda \boldsymbol{I})\,\boldsymbol{x} = 0 \tag{3.11.4}$$

This is reasonably straightforward since the matrix has such a simple tri-diagonal form and can be easily done using a standard algorithm such as LU decomposition [52] or any of the other methods for tri-diagonal eigensystems given in chapter 11 of the same reference. We then have to convert the eigenvectors from the basis $\boldsymbol{v}_i$ of N Lanczos vectors, back to the original basis of SDs.

$$\boldsymbol{V}_H = \sum_{i=1}^{N} x_i \boldsymbol{v}_i \tag{3.11.5}$$

where $\boldsymbol{x}$ is the eigenvector of $\boldsymbol{H}'$ and $\boldsymbol{v}_i$ are the Lanczos vectors.

3.12 Treatment of Ghost States

> "It doesn't matter what colour the cat is, as long as it catches mice".
>
> *Deng Xiao-Ping*

When we introduced the use of Slater Determinants in section 3.3.3 we mentioned the use of the colour condition $R \leq G \leq B$ to take advantage of the colour symmetry of the problem. We now give more details of how this is put into effect.

Suppose we have an SD, $|\, i \,\rangle$, in the basis, with multiplicity m. In the fully expanded basis we could represent it and its ghosts by the set

$$|\, i \,\rangle, |\, i+1 \,\rangle, |\, i+2 \,\rangle, \ldots, |\, i+m-2 \,\rangle, |\, i+m-1 \,\rangle \tag{3.12.1}$$

Since these states differ only in the ordering of the colours, the colour invariance of the Hamiltonian implies

$$\langle i \mid H \mid i \rangle = \langle i+1 \mid H \mid i+1 \rangle = \langle i+2 \mid H \mid i+2 \rangle = \ldots \tag{3.12.2}$$

In the reduced basis, only $\mid i \rangle$, itself, needs to appear explicitly, because, in a colourless state, each of these basis vectors will have exactly the same coefficient, which only needs to be found once. To see this, first note that any colourless state must be unaltered by permutation operators acting on the colour indices r, g, b (*i.e.* changing RGB into GRB, BRG, etc.). The set of a basis state and its ghosts is closed under operations of this kind—they merely alter the order of the basis. Thus any eigenstates found with the full basis which did not have the same coefficients for a state and its ghosts could not be colourless and would be unphysical.

With the use of the colour condition, a state and its ghosts are automatically constrained to have the same coefficients in the full basis. So this approach of leaving out the ghost states has the additional benefit of throwing away these unphysical, coloured states right from the beginning without losing any of the colourless states which we are interested in.

Now suppose we have two such SDs in the full basis, $\mid i \rangle$ and $\mid j \rangle$, with multiplicities m_i, m_j and coefficients a_i and a_j, respectively in some state vector. Suppose further that these same SDs appear in the reduced basis as $\mid i' \rangle$ and $\mid j' \rangle$, respectively [6].

If we act on this state vector with a matrix operator H, then summing up all transitions which transform $\mid j \rangle$ to $\mid i \rangle$ will contribute the amount $\langle i \mid H \mid j \rangle a_j$ to the i^{th} element of the new vector.

In starting with the SD $\mid j \rangle$, there will be additional transitions which create the ghost states of $\mid i \rangle$ mentioned in equation 3.12.1. Consider $\langle i+1 \mid H \mid j \rangle$. If we are using the reduced basis, there is no element in our state vector corresponding to $\mid i+1 \rangle$ so this contribution cannot appear explicitly. However, in the full basis, there would have been an identical matrix element $\langle j \mid H \mid i+1 \rangle$ contributing to the j^{th} element of the new state vector and it is essential that this is taken into account. This is done by adding all such transitions to the corresponding matrix element of the state of which $\mid i+1 \rangle$ is a ghost—$\langle j' \mid H \mid i' \rangle$ in this case. So, if we take account of the symmetry between a state and its ghosts, we end up with upper-triangular elements in

[6] The states are exactly the same, but their positions will obviously be different in the two bases, hence the different indices $i' and j'$ for the reduced basis.

the reduced matrix which have the form

$$\langle i' \mid H \mid j' \rangle = m_j \langle i \mid H \mid j \rangle \tag{3.12.3}$$

because of these additional transitions. This now means that the matrix we have is no-longer symmetric.

This is a nuisance as more complex diagonalisation techniques would be required, but fortunately symmetry can be restored. We multiply each coefficient a_i in the reduced basis by $\sqrt{m_i}$ and each element $\langle i' \mid H \mid j' \rangle$ by $\sqrt{m_i}/\sqrt{m_j}$. With these weighting factors, the symmetry of the matrix is restored and the calculation remains entirely consistent with the full basis version. Furthermore, since a coefficient c_i in the reduced basis replaces m_i coefficients d_i ($c_i = \sqrt{m_i} d_i$) in the extended basis, there is no need for special scalar product or normalisation routines. We simply add up the products of corresponding elements' amplitudes.

3.12.1 Reduced Basis Example

The above description may seem a bit obscure, so we consider a very basic example. Suppose we have have a reduced basis, consisting of two states with multiplicities 1 and 3, and coefficients a_1 and b_1 in some vector. The situation in the full basis would be

$$\begin{bmatrix} H_{11} & H_{12} & H_{12} & H_{12} \\ H_{12} & H_{22} & H_{23} & H_{23} \\ H_{12} & H_{23} & H_{22} & H_{23} \\ H_{12} & H_{23} & H_{23} & H_{22} \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ b_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} a_2 \\ b_2 \\ b_2 \\ b_2 \end{bmatrix} \tag{3.12.4}$$

where

$$\begin{aligned} a_2 &= H_{11} a_1 + 3 H_{12} b_1 \\ b_2 &= H_{12} a_1 + (H_{22} + 2 H_{23}) b_1 \end{aligned}$$

The situation in the reduced basis is then

$$\begin{bmatrix} H_{11} & 3H_{12} \\ H_{12} & H_{22} + 2H_{23} \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \tag{3.12.5}$$

Multiplying the vector elements by $\sqrt{m}$ and matrix elements H_{ij} by $\sqrt{m_i}/\sqrt{m_j}$ gives

$$\begin{bmatrix} H_{11} & \sqrt{3}H_{12} \\ \sqrt{3}H_{12} & H_{22}+2H_{23} \end{bmatrix} \begin{bmatrix} a_1 \\ \sqrt{3}b_1 \end{bmatrix} = \begin{bmatrix} a_2 \\ \sqrt{3}b_2 \end{bmatrix} \tag{3.12.6}$$

So diagonalising the version of the reduced matrix weighted in this way will yield eigenvectors with the appropriate factors. The dot products and normalisation of these vectors are automatically the same as those in the full basis.

3.13 Kinetic Energy Operator

The total kinetic energy operator, T, is the conventional kinetic energy minus the kinetic energy of the centre of mass of the system:

$$\begin{aligned} T &= \sum_i \frac{p_i^2}{2m_i} - \frac{\left(\sum_i \boldsymbol{p}_i\right)^2}{2M} \\ &= \sum_i \frac{p_i^2}{2m_i} - \frac{\sum_i p_i^2 + 2\sum_{i<j} \boldsymbol{p}_i \cdot \boldsymbol{p}_j}{2M} \\ &= \sum_i \left(\frac{1}{2m_i} - \frac{1}{2M}\right) p_i^2 - \frac{\sum_{i<j} \boldsymbol{p}_i \cdot \boldsymbol{p}_j}{M} \end{aligned} \tag{3.13.1}$$

So we require single particle matrix elements of p^2 and two body matrix elements of the form $\boldsymbol{p}_1 \cdot \boldsymbol{p}_2$, in addition to the masses for each orbit. These are calculated in Section A.2. Armed with these, this operator can easily be added into the Hamiltonian during its assembly.

3.14 Expectation Values of J^2, T^2 and C^2

These operators all commute with the Hamiltonian, so have simultaneous eigenfunctions. Once we have found an eigenstate of the Hamiltonian, we can calculate the expectation values of these operators and use the corresponding quantum numbers, eg. the spin quantum number, j,

$$j(j+1) = \langle \boldsymbol{J}^2 \rangle \tag{3.14.1}$$

to help identify the state[7]. They also provide a good test of convergence, since they yield sharp half-integer or integer values for fully converged states. Using $\boldsymbol{J}^2$ as an example, the operators have the form

$$\begin{aligned} \boldsymbol{J}^2 &= \left(\sum_{i=1}^{A} \boldsymbol{J}_i\right)^2 \\ &= \sum_{i=1}^{A} \boldsymbol{J}_i^2 + 2\sum_{i<j} \boldsymbol{J}_i \cdot \boldsymbol{J}_j \end{aligned} \tag{3.14.2}$$

In our model, the one body part of these operators is diagonal and each term evaluates to a constant. For $\boldsymbol{J}^2$ and $\boldsymbol{T}^2$, the value is 3/4 per particle, since the quarks have spin and isospin 1/2. For $\boldsymbol{C}^2$, the value is 16/3 per particle, which is the expectation value of the one-body colour operator, $\boldsymbol{\lambda}^2$. So we have

$$\boldsymbol{J}^2 = \frac{3}{4}A + 2\sum_{i<j} \boldsymbol{J}_i \cdot \boldsymbol{J}_j \tag{3.14.3}$$

$$\boldsymbol{T}^2 = \frac{3}{4}A + 2\sum_{i<j} \boldsymbol{T}_i \cdot \boldsymbol{T}_j \tag{3.14.4}$$

$$\boldsymbol{C}^2 = \frac{16}{3}A + 2\sum_{i<j} \boldsymbol{\lambda}_i \cdot \boldsymbol{\lambda}_j \tag{3.14.5}$$

The two body matrix elements of $\boldsymbol{J}_i \cdot \boldsymbol{J}_j$, $\boldsymbol{T}_i \cdot \boldsymbol{T}_j$ and $\boldsymbol{\lambda}_i \cdot \boldsymbol{\lambda}_j$ are worked out along with those required for the Hamiltonian. It would be possible to obtain eigenvalues for the operators in the same way as we did above, but since we have already found the eigenstates, it is a much simpler matter to evaluate them using a *density matrix.*

3.15 Density Matrix

The use of a density matrix greatly speeds up the calculation of expectation values for individual one and two-body operators. If we have an eigenstate, $|\,\Phi\,\rangle$ in an N dimensional basis,

$$|\,\phi\,\rangle = \sum_{i=1}^{N} C_i\,|\,i\,\rangle \tag{3.15.1}$$

[7]Using units where $\hbar = 1$

the density matrix, for our purposes, can then be defined by the operator

$$\rho = \mid \phi \rangle \langle \phi \mid \tag{3.15.2}$$

and its matrix elements are given by

$$\begin{aligned} \langle i \mid \rho \mid j \rangle &= \langle i \mid \phi \rangle \langle \phi \mid j \rangle \\ &= C_i C_j^* \end{aligned} \tag{3.15.3}$$

A two-body operator O is represented by

$$O = \sum O_{\mu\nu\kappa\lambda} a_\mu^\dagger a_\nu^\dagger a_\kappa a_\lambda \tag{3.15.4}$$

If we want to find the expectation value of O in the state $\mid \phi \rangle$, it is given by

$$\begin{aligned} \langle O \rangle &= \frac{\langle \phi \mid O \mid \phi \rangle}{\langle \phi \mid \phi \rangle} \\ &= \sum_{i,j} C_i C_j^* \langle i \mid O \mid j \rangle \end{aligned} \tag{3.15.5}$$

$\langle \phi \mid \phi \rangle = 1$ since our states are all normalised to unity. In our calculation, the two body matrix elements are stored for each operator. So, in the above equation,

$$\langle i \mid O \mid j \rangle = \sum_{i,j} C_i C_j \sum_{\mu\nu\kappa\lambda} O_{\mu\nu\kappa\lambda} \langle i \mid a_\mu^\dagger a_\nu^\dagger a_\kappa a_\lambda \mid j \rangle \tag{3.15.6}$$

In practice, the actual evaluation is carried out a bit differently. It will be remembered that our system involves storing a list of transitions between basis states and a pointer to the matrix elements associated with these transitions (Section 3.4). With this in mind, we scan through this list, summing up all the factors $C_i C_j$ for the listed states i, j and storing them in an array, indexed by the corresponding matrix element labels.

Then the expectation value for any particular operator can be found by summing its two body matrix elements, weighted with the factors from this array. In addition, a density matrix of factors $C_i C_j / m$ is usually stored, where m is the mass of the state $\mid j \rangle$. This is used when calculating the expectation value of the kinetic energy. It has to be done this way because the number of particles in the basis states can vary, so the mass is not constant.

3.16 Projection Operators

These are used to extract information about the quantum numbers of the individual clusters in the model. When the separation is large enough, the eigenstates split into two 3-quark clusters and we can identify them as separate particles.

Projection operator matrix elements are calculated for particle number, spin, isospin and colour and their expectation values are found using the density matrix as described above. It is sufficient to do this for the f_+ cluster—the f_- can then be identified by comparison with the overall quantum numbers of the system.

3.16.1 Number Projection

The aim is to find the number of particles in the positive cluster. Let $a^\dagger$ be a creation operator for the state f_+, centred on $z = Z/2$, then if we use the operator $n_+ = a_+^\dagger a_+$

$$\begin{aligned} n_+ \mid f_+ \rangle &= \mid f_+ \rangle \\ n_+ \mid f_- \rangle &= 0 \end{aligned} \tag{3.16.1}$$

and

$$\langle \phi_\mu \mid n_+ \mid \phi_\nu \rangle = \frac{1}{2} \tag{3.16.2}$$

$$\langle \phi_\mu \mid n_+^2 \mid \phi_\nu \rangle = \frac{1}{2} \tag{3.16.3}$$

$$\langle \phi_\mu \phi_\nu \mid n_{+i} n_{+j} \mid \phi_\kappa \phi_\lambda \rangle = \frac{1}{4} \tag{3.16.4}$$

Unfortunately, if the operator $\sum_{orbits} n_{+i}$ is applied to any six particle state it will have the eigenvalue three, which doesn't provide any useful information. Instead we can use the square of this operator,

$$\mathcal{N}_+^2 = \sum_i n_{+i}^2 + 2 \sum_{i<j} n_{+i} n_{+j} \tag{3.16.5}$$

When calculating the expectation value of this, the one-body term evaluates to $\langle A \rangle / 2$. Matrix elements for the two body term are zero unless spin, isospin,

1.	$m_{s1} = m_{s3}$,	$m_{s2} = m_{s4}$
2.	$m_{t1} = m_{t3}$,	$m_{t2} = m_{t4}$
3.	$p_1 = p_3$,	$p_2 = p_4$
4.	$c_1 = c_3$,	$c_2 = c_4$

Table 3.1: Quantum number conservation conditions for projection operators

intrinsic parity and colour of the two particles are individually conserved as shown in table 3.1.

The spin ($\mathcal{S}_+^2$), isospin ($\mathcal{T}_+^2$) and colour ($\mathcal{C}_+^2$) operators all have a form similar to that for number projection

$$\begin{aligned} O_+^2 &= \left(\sum_i n_{+i} O_i\right)^2 \\ &= \sum_i n_{+i}^2 O_i^2 + 2\sum_{i<j} n_{+i} n_{+j} O_i O_j \end{aligned} \tag{3.16.6}$$

The n_+ factors are included to ensure that the operators only act on particles in the f_+ cluster.

For the spin case, the one body term $n_{+i}^2 s_i^2$ is equal to 3/8. The two body terms are zero unless conditions 2–4 above are satisfied. In this case

$$\langle n_{+i} n_{+j} \boldsymbol{s}_i \cdot \boldsymbol{s}_j \rangle = \frac{1}{4} \langle \boldsymbol{s}_i \cdot \boldsymbol{s}_j \rangle \tag{3.16.7}$$

For isospin and colour projection, the situation is exactly the same except that the conditions 1,3,4 for isospin, and 1–3 for colour, must be satisfied for the terms to be non-zero.

3.16.2 Eigenvalues of Projection Operators

With a system of six particles, there are four possible ways of splitting the particles between two clusters. These are shown in table 3.2, along with the corresponding eigenvalues of $\mathcal{N}_+^2$. The model is symmetric in that a separation into two clusters of 6 particles and 0 particles has equal probability of having 6 particles centred on $z = +Z/2$ and 0 centred on $z = -Z/2$, or 0 particles at $z = +Z/2$ and 6 at $z = -Z/2$. The eigenvalue of $\mathcal{N}_+^2$ is then $(6^2 + 0^2)/2 = 18$,

Particles in Cluster 1	Particles in Cluster 2	$\langle \mathcal{N}_+^2 \rangle$
6	0	18
5	1	13
4	2	10
3	3	9

Table 3.2: Eigenvalues of $\mathcal{N}_+^2$

as shown in the table. The spin, isospin and colour of the f_+ cluster are found from the operators in the same way as for the full system, taking the factor of 1/4 into account for the two-body matrix elements. Alternatively, one can

Cluster 1 Spin (Isospin)	Cluster 2 Spin (Isospin)	Eigenvalue of $\mathcal{S}_+^2(\mathcal{T}_+^2)$
0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0.75
1	0	1
1	1	2
$\frac{1}{2}$	$\frac{3}{2}$	2.25
$\frac{3}{2}$	$\frac{3}{2}$	3.75
2	1	4

Table 3.3: Spin and isospin projection eigenvalues

look directly at the eigenvalues of the projection operators; $\mathcal{C}_+^2$ will be zero in situations where each individual cluster is colourless and non-zero otherwise. It should be noted that these values only apply at large cluster separations. When the cluster wavefunctions overlap, the operators will not have clearly defined eigenvalues but still give information about the structure of the states. The spin and isospin projection eigenvalues are shown in table 3.3.

Chapter 4

Potentials and their Matrix Elements

4.1 Introduction

This chapter contains a more in depth discussion of the actual interaction potentials used in the Hamiltonian and the calculation of matrix elements for the operators they contain. More detail is also given of calculations for additional operators used in the model, many of which were introduced in Chapter 3.

As mentioned previously, the interaction has two main constituents: a two-body interaction describing the first three diagrams of figure 1.2 and a number non-conserving part describing the creation of $q\bar{q}$ pairs (diagrams 4 and 5).

4.2 Quark-Quark Interaction Potential

This provides a QCD-motivated, non-relativistic approximation to one-gluon exchange, analogous to the Breit-Fermi interaction arising from one-photon exchange in QED [53]. It also contains the confining potential. The exact form we use is taken from the papers by Hecht and Fujiwara on their studies of the NN interaction [36, 54, 55]. The contribution to the Hamiltonian (*ie* $V(r_{ij})$ in equation 3.1.1) is:

$$\alpha_s \hbar c \frac{(\boldsymbol{\lambda}_i \cdot \boldsymbol{\lambda}_j)}{4} \left\{ \frac{1}{r_{ij}} - \frac{\pi \hbar^2}{m^2 c^2} \underbrace{\left[1 + \frac{2}{3}(\boldsymbol{\sigma}_i \cdot \boldsymbol{\sigma}_j)\right] \delta^3(\boldsymbol{r}_{ij})}_{\text{Colour Magnetic Term}} \right\} - \underbrace{(\boldsymbol{\lambda}_i \cdot \boldsymbol{\lambda}_j)\, a_c r_{ij}^2}_{\text{Confinement Term}} \quad (4.2.1)$$

The λ's are the Gell-Mann matrices, the generators of SU(3) colour (see section 4.7). The $\boldsymbol{\sigma}_i$'s are the spin operators for the particles, with components $\sigma_x, \sigma_y, \sigma_z$ – the Pauli spin matrices. α_s is the strong coupling constant, m the quark mass, a_c the strength of the confining potential. $\boldsymbol{r}_{ij} = \boldsymbol{x}_i - \boldsymbol{x}_j$ is the vector spacing of the two particles.

In addition to the colour-magnetic interaction term, there are several other corrections to the $1/r$ gluon-exchange potential which have been left out of equation 4.2.1. The spin-orbit interaction does not appear since we are considering S-states only. In any case, it is often neglected in many works because, experimentally, the spin-orbit effects in quark systems appear to be heavily suppressed; they are much smaller than the spin-spin interaction effects. This has not been adequately explained, although some bag model calculations for excited baryon states have shown a cancellation of the spin-orbit force [56]. There is also a tensor interaction term which appears in the Hamiltonian, but it has been shown to be of little importance in calculations of the baryon-baryon interaction [25].

The confinement term shown here is quadratic. The effect of different types of confinement potential have been investigated in other works [57] and do not alter the results appreciably. In the next chapter, we will use a linear confinement potential with some of the chosen parameter sets.

4.2.1 Evaluation of Spatial Matrix Elements

We are interested in finding matrix elements of two-body operators $1/r$, r, r^2 and $\delta^3(\boldsymbol{r})$ between two-particle states made up from the f_+ and f_- functions previously intriduced in section 3.2. Full details for working out the integrals can be found in appendix A. The quantities corresponding to these terms,

$$I_{term} = \langle\, f_{\mu_1} f_{\mu_2} \mid term \mid f_{\nu_1} f_{\nu_2} \,\rangle \quad (4.2.2)$$

which we have to evaluate in the program code, can be summarised as

$$I_1 = \frac{\sqrt{2}}{\gamma C^2} e^{\frac{\gamma^2 C^2}{4}} \mathrm{erf}\left(\frac{\gamma C}{2}\right) \tag{4.2.3}$$

$$I_2 = \frac{1}{\sqrt{2}\gamma}\left(2+\gamma^2 C^2\right) e^{\frac{\gamma^2 C^2}{4}} \mathrm{erf}\left(\frac{\gamma C}{2}\right) + \sqrt{\frac{2}{\pi}} C \tag{4.2.4}$$

$$I_2 = 2C^2\left(\frac{3}{2}+\frac{\gamma^2 c^2}{4}\right) e^{\frac{\gamma^2 C^2}{4}} \tag{4.2.5}$$

$$I_4 = \frac{1}{(2\pi C^2)^{\frac{3}{2}}} \tag{4.2.6}$$

where

$$\gamma = \frac{Z}{2\sqrt{2}C^2}\left(\mu_1+\nu_1-\mu_2-\nu_2\right) \tag{4.2.7}$$

and the μs and νs take the values ± 1.

4.3 $q\bar{q}$ Creation Interaction

As we have already discussed, our aim is to investigate the effects of including $q\bar{q}$ excitations in our $6q$ model so we need some sort of potential to describe diagrams four and five of figure 1.2, depicting the processes $q \rightarrow qq\bar{q}$ and $\bar{q} \rightarrow \bar{q}\bar{q}q$. Again we follow [36] and use the potential derived by Yu and Zhang in their work on nucleon-meson coupling constants [58]. It takes the form

$$v_{12} = \frac{i\alpha_s \hbar^2}{4m}\frac{(\boldsymbol{\lambda}_i \cdot \boldsymbol{\lambda}_j)}{4}\left\{-2\frac{\boldsymbol{\sigma}_2 \cdot \boldsymbol{r}}{r^3} + i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)\cdot\frac{\boldsymbol{r}}{r^3} + 2\frac{\boldsymbol{\sigma}_2 \cdot \boldsymbol{\nabla}_1}{r}\right\} \tag{4.3.1}$$

4.3.1 Evaluation of $1 \rightarrow 3$ Matrix Elements

Although the $q\bar{q}$ creation terms involve transitions in which one particle is destroyed and three are created, it is possible to evaluate the matrix element as a two-body one by converting the odd particle (usually the antiquark) into its conjugate [36], as illustrated in figure 4.1.

Suppose we have $\alpha \rightarrow \beta\gamma\delta$, and β is the odd particle to which we apply the time reversal operator. If the state $|\,\beta\,\rangle$ has quantum numbers m_s and m_t,

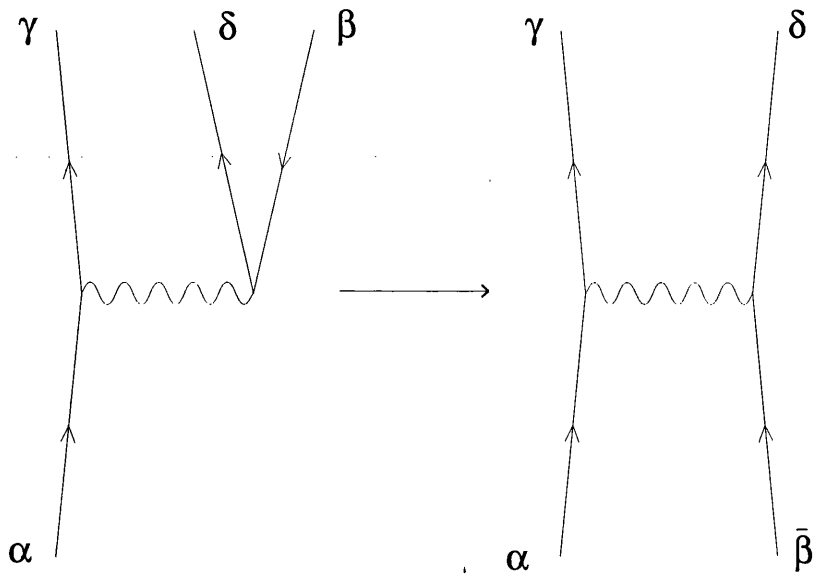

Figure 4.1: Conjugation of the odd particle

then the conjugate state $\mid \bar{\beta} \rangle$ will have $-m_s$ and $-m_t$, and will have opposite intrinsic parity to $\mid \beta \rangle$. In addition, there is a phase factor

$$(-1)^{1+m_s+m_f} \tag{4.3.2}$$

which must be taken into account [36]. We can then proceed to work out the matrix elements in the form

$$\langle \gamma\delta \mid v_{12} \mid \alpha\bar{\beta} \rangle \tag{4.3.3}$$

4.3.2 Spatial Matrix Elements

In this case the operators are $\boldsymbol{r}/r^3$ and $\boldsymbol{\nabla}_1/r$. Again full details are given in appendix A. The matrix elements, $\langle f_{\mu_1} f_{\mu_2} \mid V(\boldsymbol{r}) \mid f_{\nu_1} f_{\nu_2} \rangle$, can be calculated from these basic quantities[1]

$$I_1 = \frac{2}{\gamma C^3} T_{\text{expl}} \left[T_{\text{exp2}} T_{\text{erf}} - \frac{1}{\sqrt{\pi}} \right] \tag{4.3.4}$$

[1] I_1 corresponds to the integral for the first term. $I_2 + I_3 + I_4$ gives the integral for the second term.

$$I_2 = -\frac{1}{2C^3}(A+B)\,T_{\text{expl}}T_{\text{exp2}}T_{\text{erf}} \tag{4.3.5}$$

$$I_3 = -\frac{1}{\gamma C^3}T_{\text{expl}}\left[\left(\frac{\gamma^2C^2}{2}-1\right)T_{\text{exp2}}T_{\text{erf}}+\frac{1}{\sqrt{\pi}}\right] \tag{4.3.6}$$

$$I_4 = \frac{\nu_1 Z}{\sqrt{2}C^3}T_{\text{expl}}T_{\text{exp2}}T_{\text{erf}} \tag{4.3.7}$$

where

$$T_{\text{expl}} = e^{\frac{1}{2C^2}\left[\frac{(A-B)^2}{2}+a^2+b^2\right]} \tag{4.3.8}$$

$$T_{\text{exp2}} = e^{\frac{\gamma^2C^2}{4}} \tag{4.3.9}$$

$$T_{\text{erf}} = \frac{1}{\gamma C}\text{erf}\left(\frac{\gamma C}{2}\right) \tag{4.3.10}$$

$$\tag{4.3.11}$$

and

$$A = \frac{1}{2\sqrt{2}}Z(\mu_1+\mu_2)\boldsymbol{e}_3 \tag{4.3.12}$$

$$a = \frac{1}{2\sqrt{2}}Z(\mu_1-\mu_2)\boldsymbol{e}_3 \tag{4.3.13}$$

$$B = \frac{1}{2\sqrt{2}}Z(\nu_1+\nu_2)\boldsymbol{e}_3 \tag{4.3.14}$$

$$b = \frac{1}{2\sqrt{2}}Z(\nu_1-\nu_2)\boldsymbol{e}_3 \tag{4.3.15}$$

4.4 Matrix Elements of $\boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2$

Calculating these spin matrix elements is straightforward. The spin operator $\boldsymbol{\sigma}$ is given by

$$\boldsymbol{\sigma} = \boldsymbol{e}_1\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}+\boldsymbol{e}_2\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}+\boldsymbol{e}_3\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \tag{4.4.1}$$

So the matrix elements between spin states $|\, m_{s1} m_{s_2} \,\rangle$ are[2]

$$\begin{aligned}
\langle \downarrow\downarrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \downarrow\downarrow \rangle &= 1 \\
\langle \downarrow\uparrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \downarrow\uparrow \rangle &= -1 \\
\langle \downarrow\uparrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \uparrow\downarrow \rangle &= 2 \\
\langle \uparrow\downarrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \downarrow\uparrow \rangle &= 2 \\
\langle \uparrow\downarrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \uparrow\downarrow \rangle &= -1 \\
\langle \uparrow\uparrow | \, \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2 \, | \uparrow\uparrow \rangle &= 1
\end{aligned} \tag{4.4.2}$$

All others are zero, as $\boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2$ is a scalar operator.

4.5 Matrix Elements of $\boldsymbol{\sigma_2}$

We adopt the approach given in [11]. It is advantageous to express the operator $\boldsymbol{\sigma}$ in terms of the components

$$\sigma_+ = -\frac{1}{\sqrt{2}} (\sigma_x + i\sigma_y) \tag{4.5.1}$$

$$\sigma_- = -\frac{1}{\sqrt{2}} (\sigma_x - i\sigma_y) \tag{4.5.2}$$

$$\sigma_0 = \sigma_z \tag{4.5.3}$$

Composing similar terms, r_+, r_- and r_0 for $\boldsymbol{r}$, we can then write the term $\boldsymbol{\sigma_2} \cdot \boldsymbol{r}$, which appears in the potential as

$$\boldsymbol{\sigma_2} \cdot \boldsymbol{r} = -\sigma_+ r_- - \sigma_- r_+ + \sigma_0 r_0 \tag{4.5.4}$$

and evaluate the spin matrix elements which occur in this equation [3]

$$\langle \, m_{s1} m_{s2} \mid \sigma_\mu \mid m_{s3} m_{s4} \, \rangle = \delta_{m_{s1}, m_{s3}} \langle \, m_{s2} \mid \sigma_\mu \mid m_{s4} \, \rangle \tag{4.5.5}$$

since the operator acts only on the second particle, and

$$\langle \, m_{s2} \mid \sigma_+ \mid m_{s4} \, \rangle = -\sqrt{2} \delta_{m_{s2},\uparrow} \delta_{m_{s4},\downarrow} \tag{4.5.6}$$

$$\langle \, m_{s2} \mid \sigma_- \mid m_{s4} \, \rangle = \sqrt{2} \delta_{m_{s2},\downarrow} \delta_{m_{s4},\uparrow} \tag{4.5.7}$$

$$\langle \, m_{s2} \mid \sigma_0 \mid m_{s4} \, \rangle = m_{s2} \delta_{m_{s2}, m_{s4}} \tag{4.5.8}$$

[2] Here we use $\uparrow$ to represent spin-up particles and $\downarrow$ to represent spin-down particles.

[3] With $\mu \in \{+, -, 0\}$.

4.6 Matrix Elements of $i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)$

The operator is

$$\begin{aligned} i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2) &= (\sigma_{1y}\sigma_{2z} - i\sigma_{1z}\sigma 2y)\,\boldsymbol{e}_x \\ &+ (\sigma_{1z}\sigma_{2x} - i\sigma_{1x}\sigma 2z)\,\boldsymbol{e}_y + (\sigma_{1x}\sigma_{2y} - i\sigma_{1y}\sigma 2x)\,\boldsymbol{e}_z \end{aligned} \tag{4.6.1}$$

Again we switch from using cartesian coordinates to the components given in equation 4.5.3. These reduce to

$$i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_+ = \sigma_{1+}\sigma_{20} - \sigma_{10}\sigma_{2+} \tag{4.6.2}$$

$$i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_0 = \sigma_{1+}\sigma_{2-} - \sigma_{1-}\sigma_{2+} \tag{4.6.3}$$

$$i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_- = \sigma_{10}\sigma_{2-} - \sigma_{1-}\sigma_{20} \tag{4.6.4}$$

Using the results from the last section we can proceed immediately to write down the matrix elements of these operators.

$$\begin{aligned}
\langle\uparrow\uparrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_+ \,|\uparrow\downarrow\rangle &= \sqrt{2} \\
\langle\uparrow\uparrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_+ \,|\downarrow\uparrow\rangle &= -\sqrt{2} \\
\langle\uparrow\downarrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_+ \,|\downarrow\downarrow\rangle &= \sqrt{2} \\
\langle\downarrow\uparrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_+ \,|\downarrow\downarrow\rangle &= -\sqrt{2} \\
\langle\uparrow\downarrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_0 \,|\downarrow\uparrow\rangle &= -2 \\
\langle\downarrow\uparrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_0 \,|\uparrow\downarrow\rangle &= 2 \\
\langle\uparrow\downarrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_- \,|\uparrow\uparrow\rangle &= \sqrt{2} \\
\langle\downarrow\uparrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_- \,|\uparrow\uparrow\rangle &= -\sqrt{2} \\
\langle\downarrow\downarrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_- \,|\uparrow\downarrow\rangle &= \sqrt{2} \\
\langle\downarrow\downarrow|\, i(\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2)_- \,|\downarrow\uparrow\rangle &= -\sqrt{2}
\end{aligned}$$

All others are zero.

4.7 Matrix Elements of $\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2$

The colour degree of freedom is described by the fundamental representation ϕ, of the group SU(3),

$$\phi = \begin{bmatrix} r \\ g \\ b \end{bmatrix} \tag{4.7.1}$$

An object in this representation has three degrees of freedom and since we are talking about colour, it is normal to label these red, green and blue (rgb). Analogous to the Pauli spin matrices of SU(2) the generators of SU(3) are the Gell-Mann matrices, λ_i

$$\lambda_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \lambda_2 = \begin{bmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \lambda_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\lambda_4 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \lambda_5 = \begin{bmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{bmatrix} \quad \lambda_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{4.7.2}$$

$$\lambda_7 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{bmatrix} \quad \lambda_8 = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

We are interested in finding matrix elements of the operator $\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2$ between two-particle states consisting of quarks and antiquarks. In group theory parlance, this means we must first look at the combination $\mathbf{3} \otimes \mathbf{3}$ of the fundamental representation. This yields 6 symmetric and three antisymmetric combinations ($\mathbf{6} \oplus \bar{\mathbf{3}}$). The symmetric ones are

$$\begin{gathered} rr \\ \tfrac{1}{\sqrt{2}}(rg + gr) \\ gg \\ \tfrac{1}{\sqrt{2}}(rb + br) \\ \tfrac{1}{\sqrt{2}}(gb + bg) \\ bb \end{gathered} \tag{4.7.3}$$

and the antisymmetric ones are

$$\begin{aligned} \bar{b} &= \tfrac{1}{\sqrt{2}}(rg - gr) \\ \bar{g} &= \tfrac{1}{\sqrt{2}}(br - rb) \\ \bar{r} &= \tfrac{1}{\sqrt{2}}(gb - bg) \end{aligned} \tag{4.7.4}$$

These states form a conjugate representation of SU(3) and can be interpreted as denoting the antiparticles of the states r, g and b. We are also interested in states which have a quark and an antiquark. These are obtained from

$$\mathbf{3} \otimes \bar{\mathbf{3}} = \mathbf{1} \oplus \mathbf{8} \tag{4.7.5}$$

The octet of states is made up of the states [59]

$$\begin{matrix} r\bar{b} & g\bar{b} & r\bar{g} \\ g\bar{r} & b\bar{g} & b\bar{r} \end{matrix} \tag{4.7.6}$$

and

$$\begin{aligned} \psi_1 &= \tfrac{1}{\sqrt{2}}\left(r\bar{r} + g\bar{g}\right) \\ \psi_2 &= \tfrac{1}{\sqrt{6}}\left(2b\bar{b} + g\bar{g} - r\bar{r}\right) \end{aligned} \tag{4.7.7}$$

The singlet state is

$$\psi_3 = \frac{1}{\sqrt{3}}\left(r\bar{r} - g\bar{g} + b\bar{b}\right) \tag{4.7.8}$$

The coefficients in these relations are just SU(3) Clebsch-Gordan coefficients. The uncoupled states $\mid r\bar{r}\,\rangle$, $\mid g\bar{g}\,\rangle$ and $\mid b\bar{b}\,\rangle$ can then be obtained from linear combinations of equations 4.7.7 and 4.7.8 via the following transformation

$$\begin{bmatrix} r\bar{r} \\ g\bar{g} \\ b\bar{b} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\ 0 & \sqrt{\frac{2}{3}} & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{bmatrix} \tag{4.7.9}$$

We now have descriptions of all the states we need in terms of SU(3) representations and we can proceed to find numerical values for our matrix elements.

The values of the SU(3) Casimir operator, $\boldsymbol{\lambda}^2$, depend on the dimension of the representation and those which concern us are shown in table 4.1. Details of their calculation can be found in [60][4]. The operator $\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2$ can be rewritten

[4]The Casimir operators are often defined as $\boldsymbol{F}^2$ where $F_i = \frac{1}{2}\lambda_i$ In this case the expectation values will be smaller by a factor of four.

Dimension	**1**	**3**	$\bar{\mathbf{3}}$	**6**	**8**
$\langle\lambda^2\rangle$	0	$\frac{16}{3}$	$\frac{16}{3}$	12	$\frac{40}{3}$

Table 4.1: Casimir Operator Values in SU(3)

as

$$\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 = \frac{1}{2}\left[(\boldsymbol{\lambda}_1 + \boldsymbol{\lambda}_2)^2 - {\boldsymbol{\lambda}_1}^2 - {\boldsymbol{\lambda}_2}^2\right] \tag{4.7.10}$$

which is a sum of Casimir operators. The $(\boldsymbol{\lambda}_1 + \boldsymbol{\lambda}_2)^2$ operator has the values in table 4.1 and the one-body operators have expectation values of $\frac{16}{3}$ because they act on the triplet representations. We can immediately proceed to evaluate equation 4.7.10.

$$\begin{aligned}
\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{1}} &= -\tfrac{16}{3} \\
\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{3}} &= -\tfrac{8}{3} \\
\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\bar{\mathbf{3}}} &= -\tfrac{8}{3} \\
\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{6}} &= \tfrac{4}{3} \\
\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{8}} &= \tfrac{2}{3}
\end{aligned} \tag{4.7.11}$$

4.7.1 qq States

For states with two quarks, when both colours are the same, we are obviously dealing with the sextet given in equation 4.7.3 and the matrix elements are $\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{6}}$. When the colours are different, we express the states as combinations of 4.7.3 and 4.7.4. So, for example,

$$\langle rg \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid rg\rangle = \frac{1}{2}\left[\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{6}} + \langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\bar{\mathbf{3}}}\right] \tag{4.7.12}$$

and

$$\langle rg \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid gr\rangle = \frac{1}{2}\left[\langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\mathbf{6}} - \langle\boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2\rangle_{\bar{\mathbf{3}}}\right] \tag{4.7.13}$$

In terms of logical colour (introduced in section 3.4.1),

$$\langle 00 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 00\rangle = \frac{4}{3} \tag{4.7.14}$$

$$\langle 01 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 01\rangle = -\frac{2}{3} \tag{4.7.15}$$

$$\langle 01 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 10\rangle = \frac{4}{3} \tag{4.7.16}$$

$$\langle 11 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 00\rangle = 0 \tag{4.7.17}$$

4.7.2 $q\bar{q}$ States

The situation with $q\bar{q}$ states with the same colour is worth mentioning explicitly. In all other cases, different states are orthogonal and the matrix elements are zero. However, in addition to $\langle\, r\bar{r} \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid r\bar{r} \,\rangle$, there are also matrix elements such as $\langle\, r\bar{r} \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid b\bar{b} \,\rangle$ which are non-zero. From equation 4.7.9, we have

$$\begin{aligned} r\bar{r} &= \frac{1}{\sqrt{2}}\psi_1 - \frac{1}{\sqrt{6}}\psi_2 + \frac{1}{\sqrt{3}}\psi_3 \\ b\bar{b} &= \sqrt{\frac{2}{3}}\psi_2 + \frac{1}{\sqrt{3}}\psi_3 \end{aligned} \tag{4.7.18}$$

So

$$\langle\, r\bar{r} \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid r\bar{r} \,\rangle = \frac{2}{3}\langle \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \rangle_{\mathbf{8}} + \frac{1}{3}\langle \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \rangle_{\mathbf{1}} \tag{4.7.19}$$

and

$$\langle\, r\bar{r} \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid b\bar{b} \,\rangle = -\frac{1}{3}\langle \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \rangle_{\mathbf{8}} + \frac{1}{3}\langle \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \rangle_{\mathbf{1}} \tag{4.7.20}$$

Again writing these in terms of logical colour, we get

$$\langle\, 00 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 00 \,\rangle = -\frac{4}{3} \tag{4.7.21}$$

$$\langle\, 11 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 00 \,\rangle = -2 \tag{4.7.22}$$

$$\langle\, 01 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 01 \,\rangle = \frac{1}{6} \tag{4.7.23}$$

$$\langle\, 10 \mid \boldsymbol{\lambda}_1 \cdot \boldsymbol{\lambda}_2 \mid 01 \,\rangle = 0 \tag{4.7.24}$$

Chapter 5

Calculations and Results

> "I have had my results for a long time: but I do not yet know how I am to arrive at them".
>
> *Karl Friedrich Gauss.*

5.1 Introduction

We are now interested in finding out what the model is actually capable of. Having written the program and calculated the matrix elements for it to use, what can it be expected to do and what can we hope to find?

To start with, some simple test calculations are described and carried out. Next, we attempt to reproduce the kind of results obtained by Storm for $6q$ calculations [28, 29] but with the interaction potential given in the previous chapter, which is different from her original one. Several sets of parameters, gleaned from the literature are used to test the robustness of the model and the stability of the results obtained.

We use projection operator techniques to analyse the states which appear in the spectra from these calculations, and study mixing effects in the states at small cluster separations. We do some investigations into the importance of the stability condition, mentioned in chapter 1 and check to what extent it is satisfied by the parameters we have used.

Finally, we move on to including $q\bar{q}$ excitation effects and see what difference these make to the NN potential. The proportion of $6q$ in the wave-

functions of the states in the new space is calculated and compared with the findings of other studies [11].

5.2 Choice of Model Parameters

> "You have a choice — death or bongo".
>
> *Popular joke — Death by Bongo*

> "While relativistic effects are important in these systems, it seems that they can for the most part be hidden in the choice of such potential model parameters as the constituent quark mass. When all the dust settles, apparently it is 'better to have the right degrees of freedom moving at the wrong speed than the wrong degrees of freedom moving at the right speed'".
>
> *Nathan Isgur* [61]

The above quotation from Nathan Isgur is included as a reminder that we are dealing with a model which is by no means perfect. In many ways we are pushing theories to their limits and possibly beyond, and it is unrealistic to expect more than qualitative results from our calculations. The choice of parameters can go some way towards "hiding" this problem but it is hardly satisfactory if all we are doing is fitting them to make the model give the best answers.

There has to be some justification of the choice of parameters but the approach taken in fitting them seems to vary quite a bit between implementations. Different groups place the emphasis on different conditions to justify the importance of their particular method.

In addition, there are certain physical constraints which must be taken into account when we pick our four parameters. An ideal set of parameters should have the following properties:

1. It should give the correct mass of the nucleon

$$E_N = 939\text{MeV} \tag{5.2.1}$$

2. It should fit the nucleon delta mass difference correctly

$$E_\Delta - E_N = 293\text{MeV} \tag{5.2.2}$$

3. The value of the oscillator parameter, C, should be consistent with the size of the nucleon (ie. about 0.8fm).

4. The constituent quark mass should be in the region of $m_N/3 = 313$MeV.

5. The "stability condition", that the mass of the nucleon should be minimised with respect to the oscillator length parameter, should be satisfied.

In practice, it isn't possible to satisfy all these simultaneously. We don't worry about the absolute values of the energies in our calculations since it is unreasonable to expect a non-relativistic treatment to give the correct masses, so we can relax condition 1 and don't expect to obtain the correct mass of the nucleon in all cases.

Although the size of the nucleon places an upper bound on the value of the oscillator parameter, it may be satisfactory to use lower values since this can be taken to represent the "core" of the nucleon. Even so, the choice is still rather arbitrary. The commonly used value of 0.6 fm corresponds to assigning about half the mean square charge radius of the proton to meson clouds surrounding the quark core [62].

The coefficient of the colour magnetic, spin-spin interaction, *ie.* the quark-gluon coupling constant, α_s, is fitted to give the $N\Delta$ mass difference.

Finally, the strength of the confining potential, a_c, is usually chosen so that the stability condition is satisfied. The emphasis placed on this condition varies between implementations and it's importance seems to depend on the details of the model. In most cases the discussion revolves around the details of the RGM [1] calculations and whether the channel is pure NN or mixing of other states is included [27]. The exact implications for our model are not immediately clear, although the original studies with our model [28, 29] did indicate that it made a substantial difference to the potentials. In this case, however, the minimisation of the nucleon energy was done by varying C directly and keeping a_c constant.

[1] Resonating group method

m_q	α_s	a_c	C
313 MeV	1.517	23.67 MeV/fm^2	0.6 fm

Table 5.1: The model parameters

To begin with, in line with our choice of potential, we use a set of parameters given in [36] and also used in [11]. They are shown in table 5.1. In choosing these, Hecht and Fujiwara don't minimise the oscillator parameter at all. They use quite a complicated fitting procedure which is described in [36]—the variation of several quantities, such as the nucleon magnetic moments, is drawn on a C versus m plot and the oscillator length and mass values are selected from this.

The value of a_c is picked to give the real nucleon mass of 939 MeV, purely for convenience, claiming that their final results are very insensitive to the choice of this parameter—a claim which is borne out by other works.

5.3 Test Calculations

The potential used by Storm is slightly different from the current one, but it is a simple matter to alter the parameters and the terms used. An obvious means of testing the program code was to try to repeat some of her calculations. The program successfully reproduced the numerical results given in her thesis [28]. Details of the potential and the parameters she used can be found there, if required. In addition, the following set of consistency checks can be made.

1. A $3q$ system is used with very small separation (0.0001fm) and only one shell (Φ_+) to simulate a one-centre calculation. This gives us the nucleon and delta masses and the splitting ($E_\Delta - E_N$).

2. With $6q$ at large separation (eg. 5.0fm) we check the energies difference of the two lowest states (10) and (12)[2]. These should be a NN and $N\Delta$ respectively so the energy difference should equal the N-Δ mass difference.

[2]Spin and isospin values (JT)

3. The NN system at large separation should be consistent with the nucleon mass. *ie* It should satisfy

$$E_{6q} = 2E_N + \frac{3}{4}\hbar\omega \tag{5.3.1}$$

The last term arises because the centre-of-mass kinetic energy is subtracted twice in the term $2E_N$, but only once in E_{6q}.

4. The third state in the $NN - 6q$ system has all the quarks in one cluster. This should have the same energy as the equivalent one centre calculation.

Doing these simple calculations with our initial parameter set gives the following results:

1. The Nucleon and Delta energies are 938.96 MeV and 1231.96 MeV respectively, giving $E_\Delta - E_N = 293.0\text{MeV}$.

2. The energies of the first and second states in a system of six quarks are 2137.097 MeV and 2430.093 MeV. The splitting is the same (292.0 MeV), which is consistent with these being a NN and a $N\Delta$ respectively.

3. The value of $\hbar\omega$ is obtained by using

$$\hbar\omega = \frac{\hbar^2}{mC^2} \tag{5.3.2}$$

With the values in table 5.1, we get

$$\hbar\omega = \frac{(\hbar c(\text{MeV fm}))^2}{m(\text{MeV})C^2(\text{fm}^2)} = 345.568\text{MeV} \tag{5.3.3}$$

giving $2E_N + \frac{3}{4}\hbar\omega = 1830.42\text{MeV}$, so this condition is satisfied.

4. With $6q$ in the Φ_+ shell, and $Z = 0.0001$ fm, the energy is 2478.935 MeV. The third state in the NN spectrum at $Z = 5.0\,\text{fm}$ has the same eigenvalue correct to the fifth decimal place.

These tests all give consistent results, providing some reassurance that the code is functioning satisfactorily.

5.4 Six Quark Calculations of Baryon-Baryon Interactions

In this section we perform calculations of interactions with a six quark system, along the same lines as those performed by Storm, but using the new potential. We would expect that this should give qualitatively the same results as she obtained, including some evidence of repulsion at low separations and little or no intermediate attraction in the NN channel.

When using the m-scheme, we specify the projections of spin and isospin, and the eigenstates obtained from the calculation will have values of J and T consistent with these. To illustrate this, the eight low lying states in the spectrum obtained from a calculation with both projections equal to zero is shown in figures 5.1 and 5.2. A similar spectrum for the same calculation with $m_s = 1$ is shown in figures 5.3 and 5.4.

The energies, $E_0, E_1 \ldots$ for these spectra are ordered such that $E_0 \leq E_1 \leq E_2 \leq \ldots$ at large separations (see table 5.2 below). The (J,T) values are shown in brackets. It can be seen that the states with $J = 0$, such as the $NN(0,1)$ state are no longer present in the second ($m_s = 1$) spectrum. This is obviously the case because $J = 0$ states can only have $m_s = 0$.

Taking a look at the full spectrum, the two states of lowest energy, E0 and E1 in figures 5.1 and 5.2 are the $^1S_0$ and $^3S_1$ nucleon-nucleon systems. Above these, there are four $N\Delta$ states, E2, E3, E4, E5, with varying amounts of repulsion in evidence at low separations. The state labelled E6 consists of 6 particles in one cluster, with deuteron quantum numbers. The higher energy state, E7, with $J = 0$, $T = 0$ is also of this form. There are eight degenerate states with the mass of a two delta particle system at large separations, some of which are incomplete at the origin, due to the scope of the calculation, which is set up to request a particular number of eigenvalues; the energies of these states at lower separations were thus too high to be found.

The other states shown in the spectra have rapidly rising energies at cluster spacings greater than 1.5 fm. These are hidden colour states—which are colourless overall, but consist of two coloured fragments. At large r values, this leads to an interaction between the clusters which take on the form of the confining potential. So this case, the energy varies as kr^2, for some con-

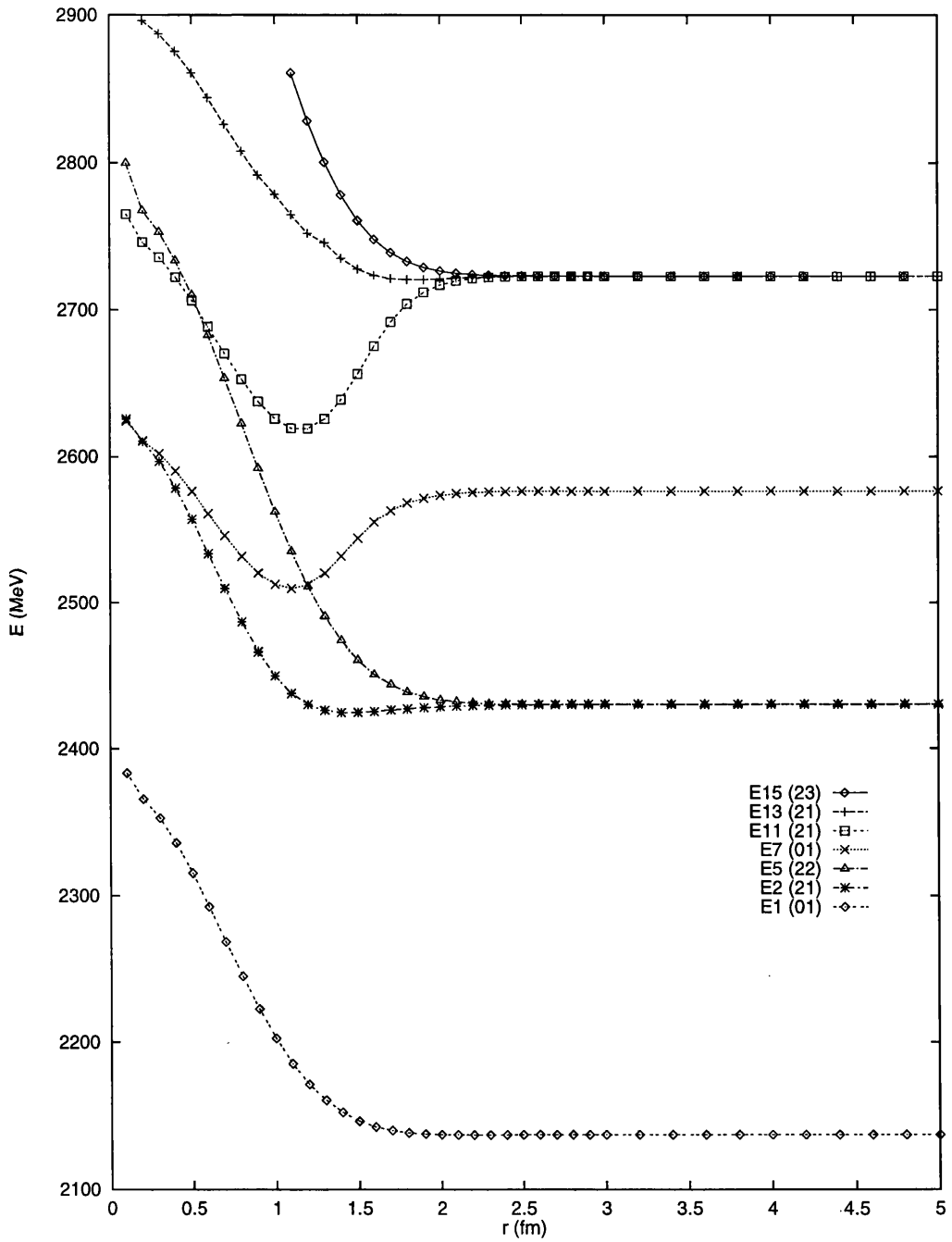

Figure 5.1: $6q$ calculation spectrum (even-$\boldsymbol{J}$ states)

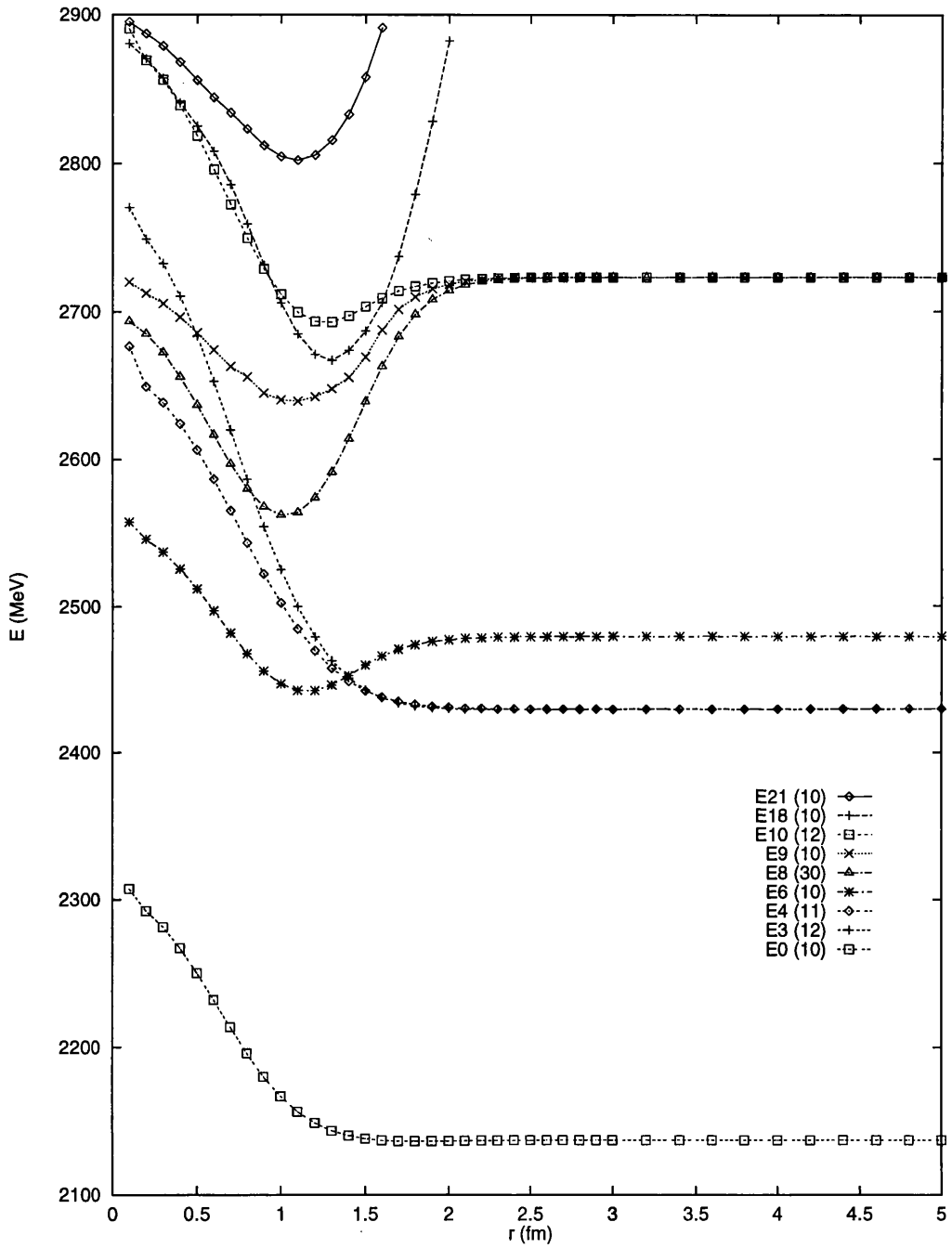

Figure 5.2: $6q$ calculation spectrum (odd-J states)

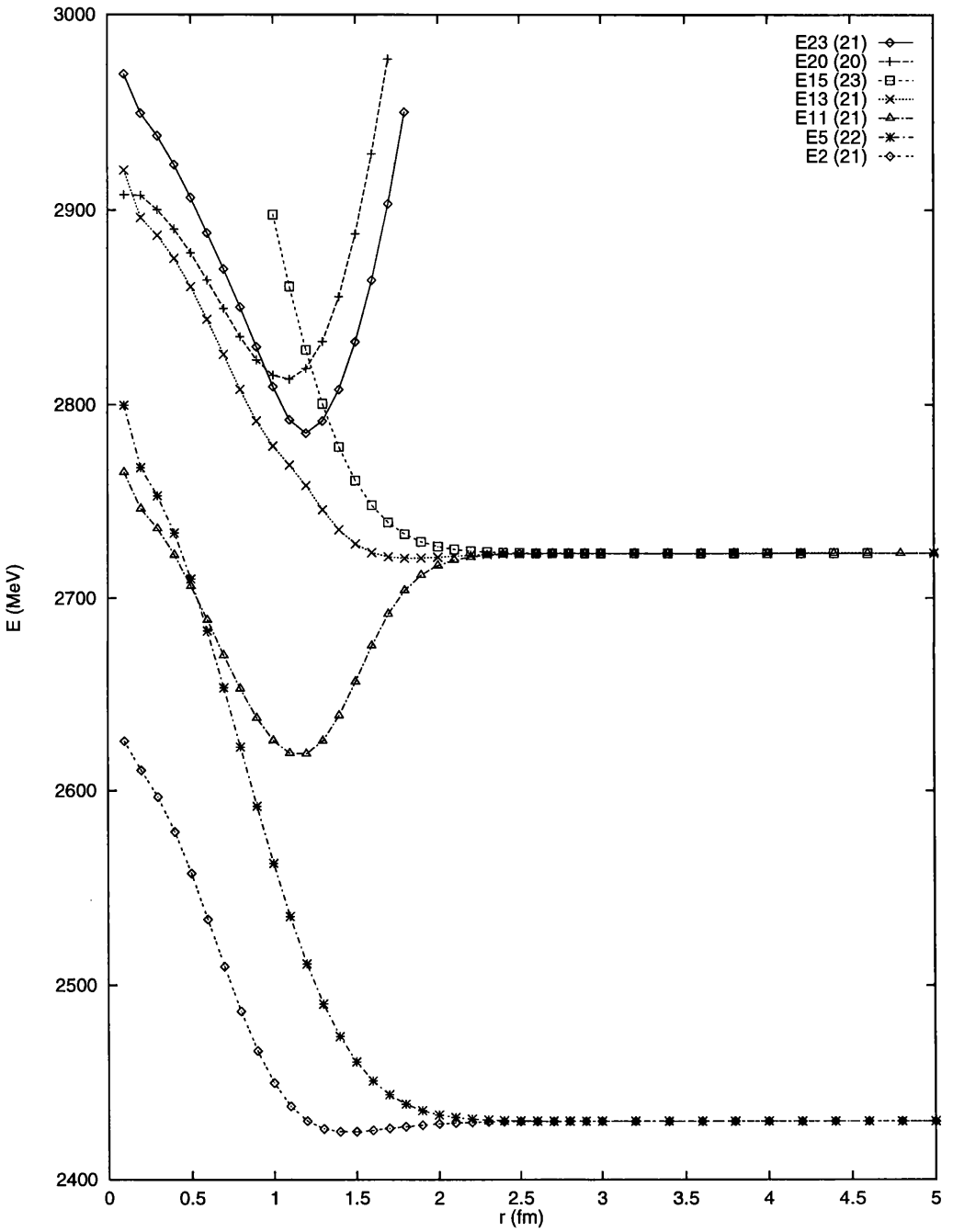

Figure 5.3: $6q$ calculation spectrum with $m_s = 1$ (even-$\boldsymbol{J}$ states)

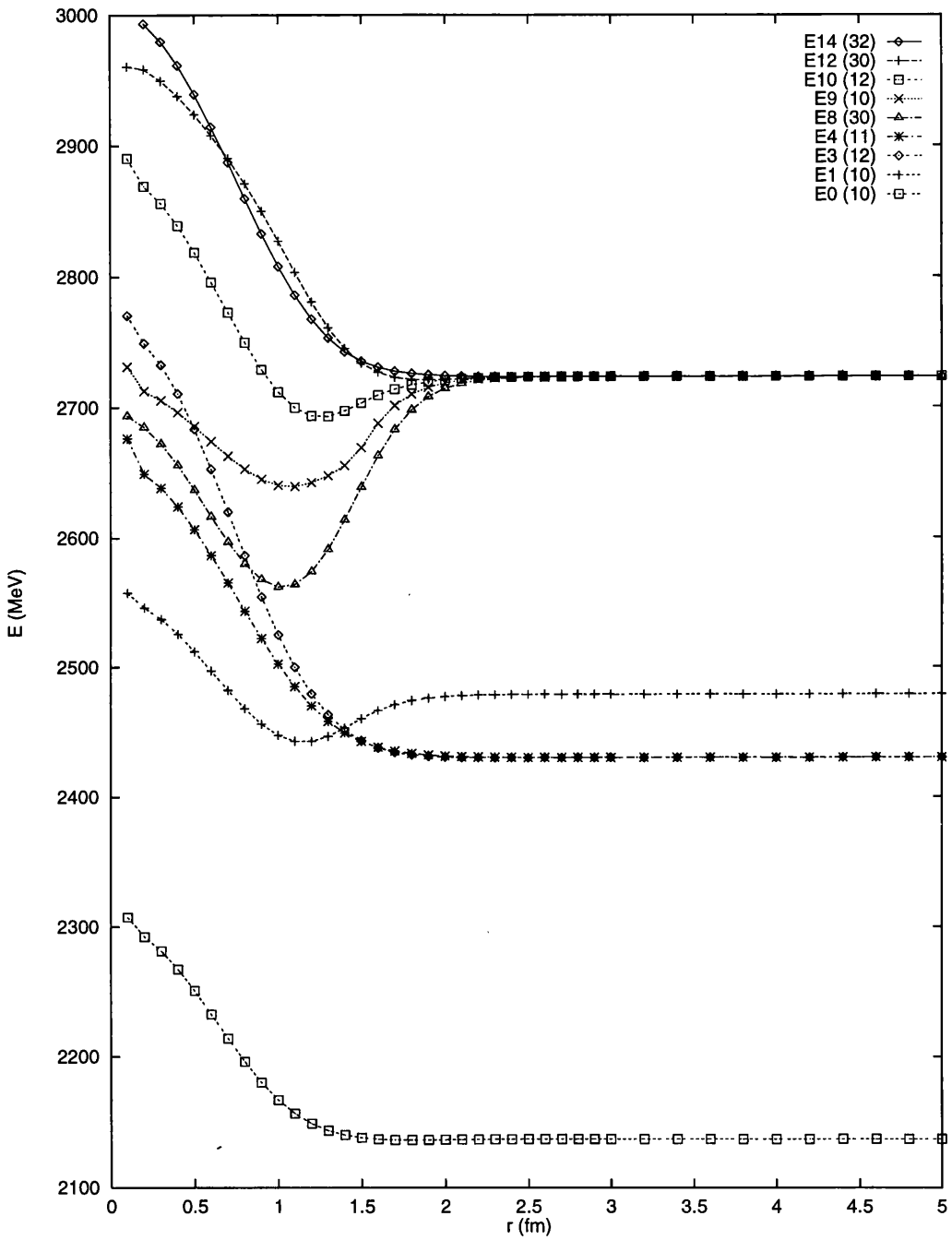

Figure 5.4: $6q$ calculation spectrum with $m_s = 1$ (odd-$\boldsymbol{J}$ states)

stant k. The importance of the hidden colour states at small separation has been stressed in other works [26, 63]. They make up a sizeable portion of the wavefunction near the origin, contributing to the repulsive core [36], but are unphysical at large separations.

5.4.1 Projection Analysis of the States

The most important states in the spectra have been identified in terms of their constituent particles (NN,$N\Delta$, etc) using the projection operators described in section 3.16. They are only clearly defined at large cluster separations. As the value of Z decreases, mixing of the states with the same quantum numbers will occur, and we can measure the degree to which this happens by recording how the amplitudes vary with separation. Table 5.2 shows the spectrum of states at a large cluster separation ($Z = 3.5fm$) and their composition. We can then work out the components of any eigenstate in terms of these states. In line with [28], we use the square of the amplitude of the component to measure the amount of mixing. Figures 5.5,5.6,5.8 show the situation for a selection of the lower lying states in the spectrum.

If we look at the first two graphs, at separations above about 2 fm, the wavefunctions are entirely NN but as the clusters are moved together, the proportion of the "pure" state in the wavefunction drops rapidly and higher energy states begin to get mixed in, including the hidden coloured states which form quite a large part of the wavefunctions near the origin.

Mixing should only occur, between states with the same quantum numbers—parity, J and T, as is confirmed in the results.

5.4.2 Alternative Parameter Sets for $6q$

To test the sensitivity of the model to variations in the parameters, we repeat the $6q$ calculations with some alternative parameters. We obtain spectra using the values given in table 5.3 which is a reproduction of Table 1 from reference [64]—one of many works on the NN interaction by Oka and Yazaki [65, 27, 25, 66, 67]. The values of a_c in this table were chosen for a

State	(J,T)	Eigenvalue	Particle Split	Composition
E0	(1,0)	2137.0964	3,3	NN
E1	(0,1)	2137.0965	3,3	NN
E2	(2,1)	2430.0926	3,3	$N\Delta$
E3	(1,2)	2430.0926	3,3	$N\Delta$
E4	(1,1)	2430.0927	3,3	$N\Delta$
E5	(2,2)	2430.0927	3,3	$N\Delta$
E6	(1,0)	2478.9353	6,0	
E7	(0,1)	2576.5907	6,0	
E8	(3,0)	2723.0887	3,3	$\Delta\Delta$
E9	(1,0)	2723.0888	3,3	$\Delta\Delta$
E10	(1,2)	2723.0889	3,3	$\Delta\Delta$
E11	(2,1)	2723.0888	6,0	
E12	(3,0)	2723.0888	6,0	
E13	(2,1)	2723.0888	3,3	$\Delta\Delta$
E14	(3,2)	2723.0889	3,3	$\Delta\Delta$
E15	(2,3)	2723.0889	3,3	$\Delta\Delta$
E16	(1,2)	2918.4195	6,0	
E17	(0,3)	3309.0811	6,0	
E18	(1,0)	4002.5998	4,2	
E19	(0,1)	4100.2651	4,2	

Table 5.2: Properties of the first 20 states at $Z = 3.5\text{f}m$

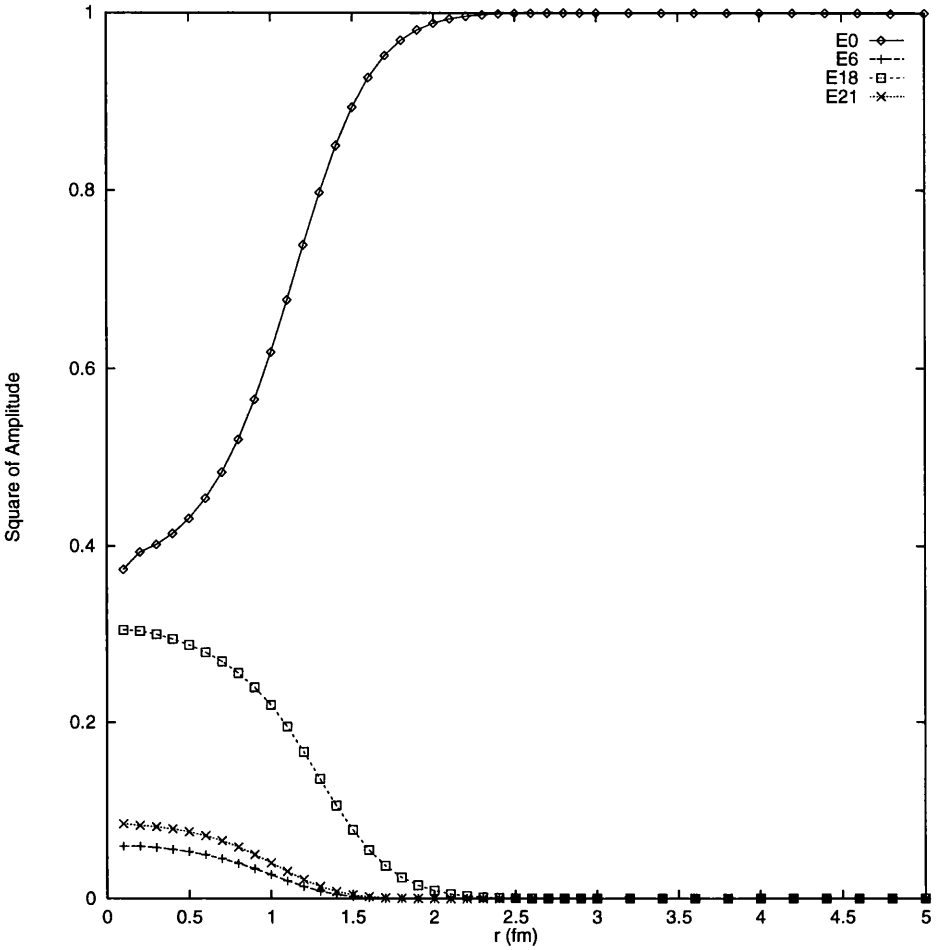

Figure 5.5: Components of first eigenstate, E0

linear confinement potential [3] but it is a simple enough matter to alter the program to accomodate this—the appropriate matrix elements were evaluated in the previous chapter. The spectra using these sets are shown in figures 5.9, 5.10, 5.11, 5.12, 5.13 and 5.14.

The results show a reasonable degree of stability between parameter sets and are qualitatively in agreement with those obtained with our original parameter set, though there are quite substantial differences in the amount short distance repulsion in the states. Obviously, the absolute energies also vary quite dramatically but, as mentioned above, we do not expect to obtain cor-

[3] We retain the symbol a_c for the linear confinement strength parameter as this is almost universally standard throughout the literature

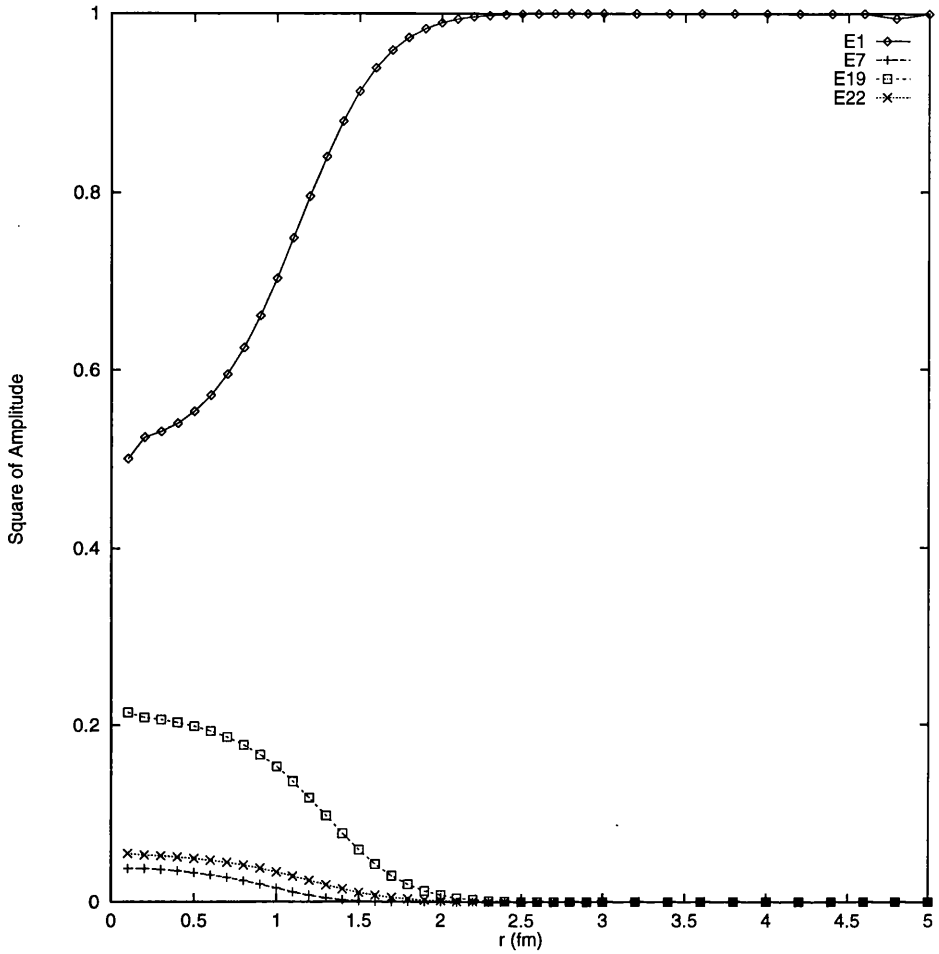

Figure 5.6: Components of second eigenstate, E1

rect masses for the system—we are interested in the variation of the energy with cluster spacing.

5.4.3 The Stability Condition

We discussed this in section 5.2 when choosing our sets of parameters. Here we investigate how well our these sets conform to the condition

$$\frac{\partial E_N}{\partial C} = 0 \tag{5.4.1}$$

and to what extent the satisfaction of the stability condition, or lack of it, affects our results. Initial impressions, from the results we have obtained so

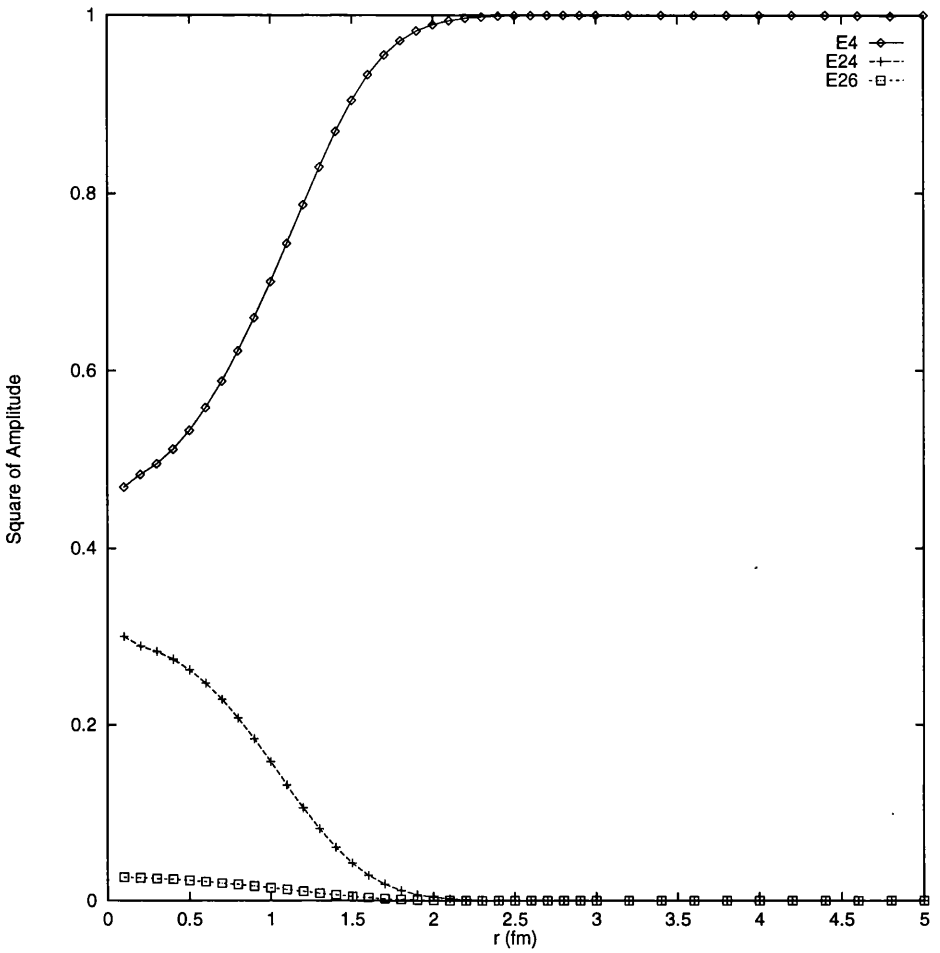

Figure 5.7: Components of fifth eigenstate, E4

far, would seem to indicate that it is not of major importance.

It will be recalled that the parameters used by Oka and Yazaki were supposedly chosen to satisfy this equation, but it isn't immediately obvious that they will do so in our model and in fact, they don't. For example, the variation of nucleon mass with C [4] for set II, used above, is shown in figure 5.15 and it can be seen that the minimum in energy occurs at 0.5 fm rather than at 0.6 fm. They have admitted [65] that the relation they use to calculate a_c neglects the presence of the OGE potential and might have to be adjusted accordingly. This was deemed unnecessary, however, since the results they obtained are so

[4] A golden section search routine [68] was used to do the minimisation

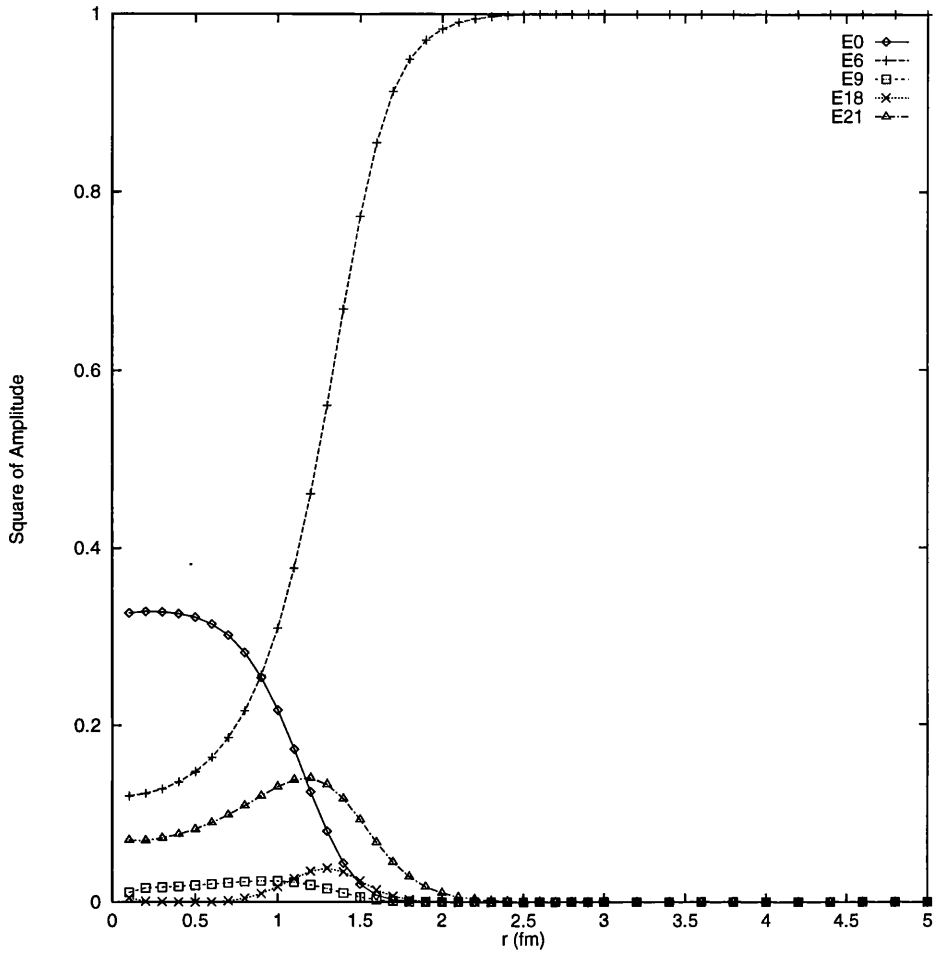

Figure 5.8: Components of seventh eigenstate, E6

insensitive to even quite substantial variations in the strength of the confining potential.

Figure 5.16 shows the graph obtained by minimising the mass of a nucleon with our original set of parameters which, in fact, only differs from Oka and Yazaki's parameter set II in the choice of a_c and the use of a quadratic confining potential. Our calculation shows the minimum occurring at 0.61 fm. The energy is minimised at $C = 0.6$ fm if we make a slight alteration to the confining strength, changing it to $a_c = 27.8$ MeV/fm^2, as shown in figure 5.17. Such a minor alteration to the parameter will not have any significant effect on the results.

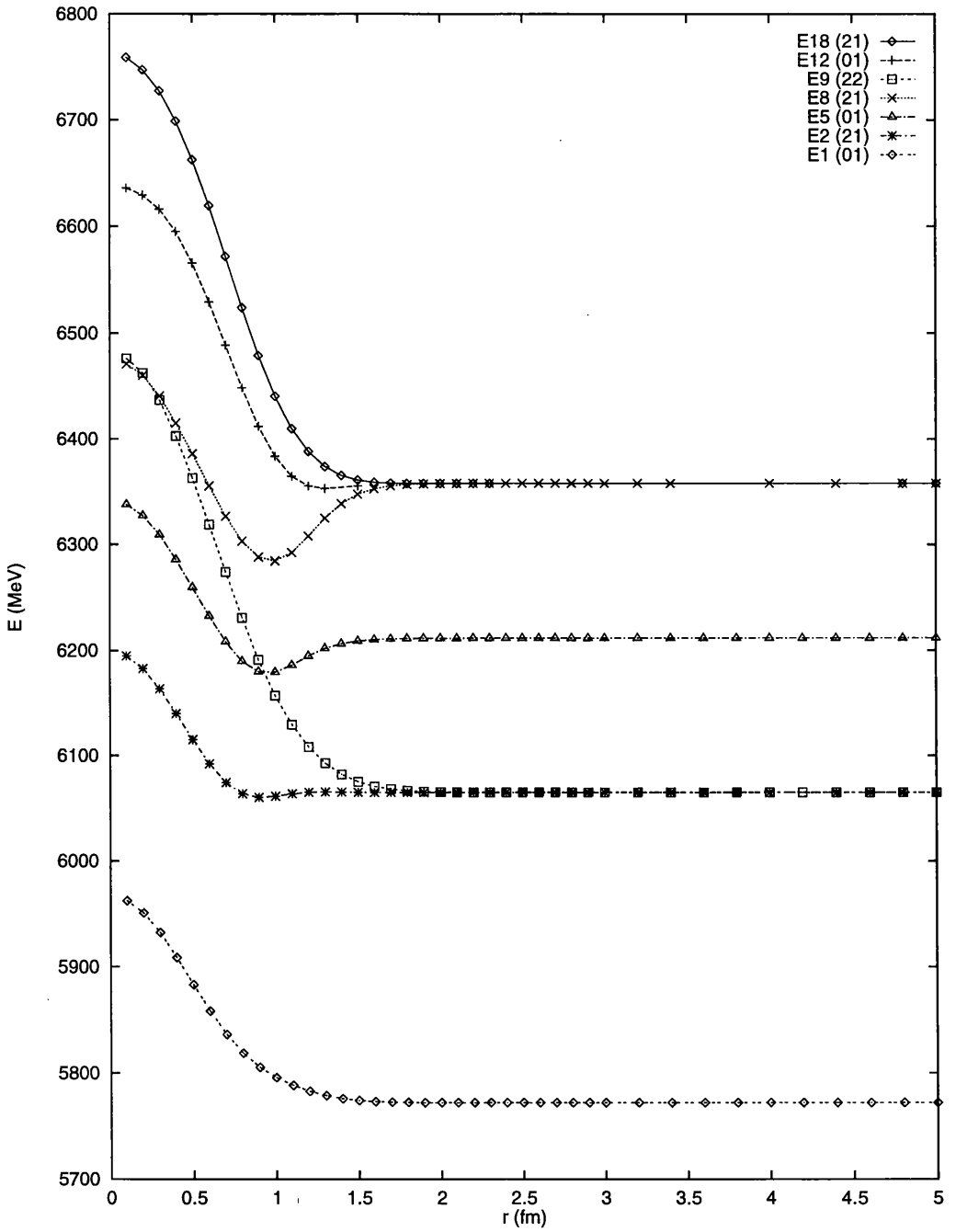

Figure 5.9: Spectrum with Oka-Yazaki Parameter set I. (even-$\boldsymbol{J}$ states)

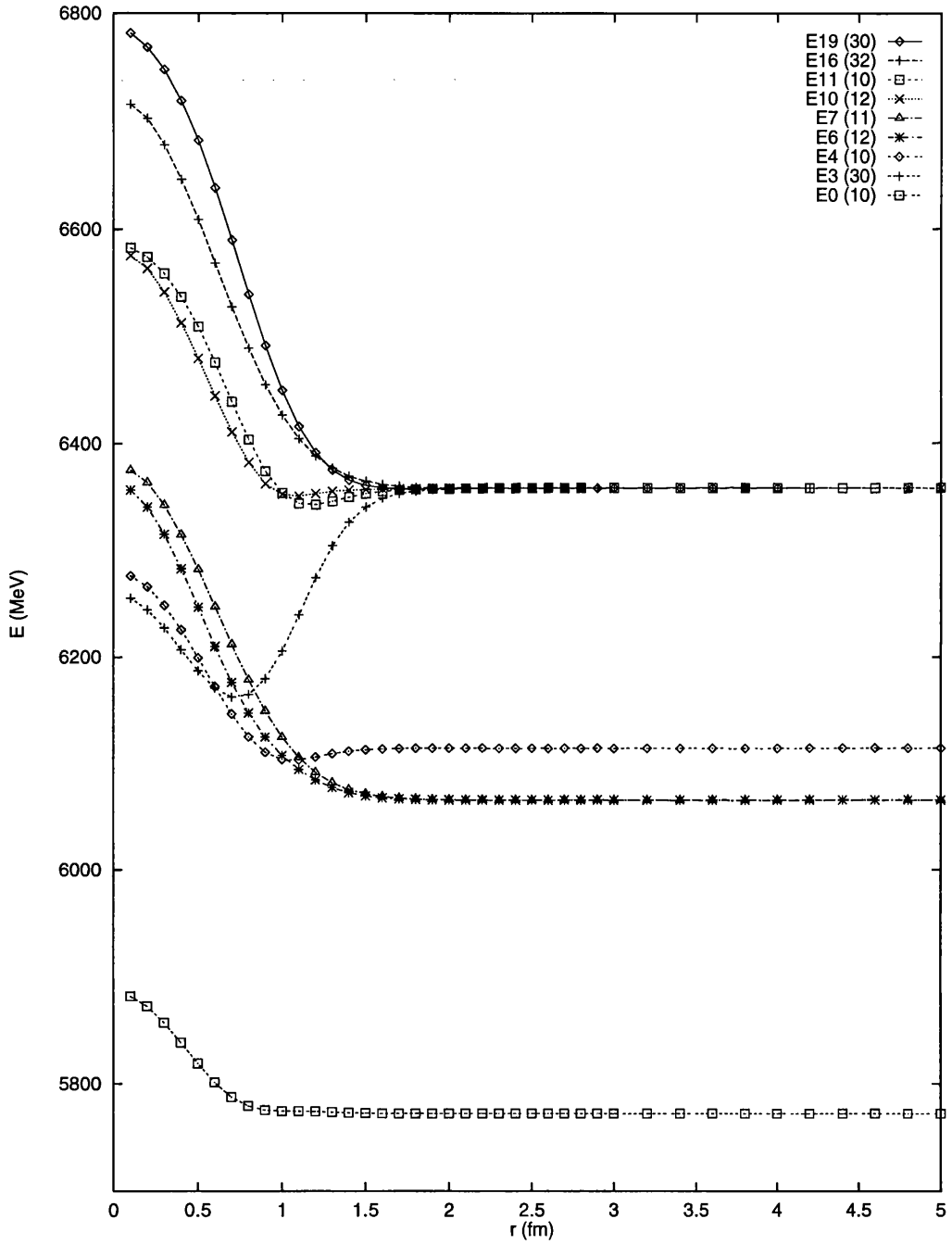

Figure 5.10: Spectrum with Oka-Yazaki Parameter set I. (odd-J states)

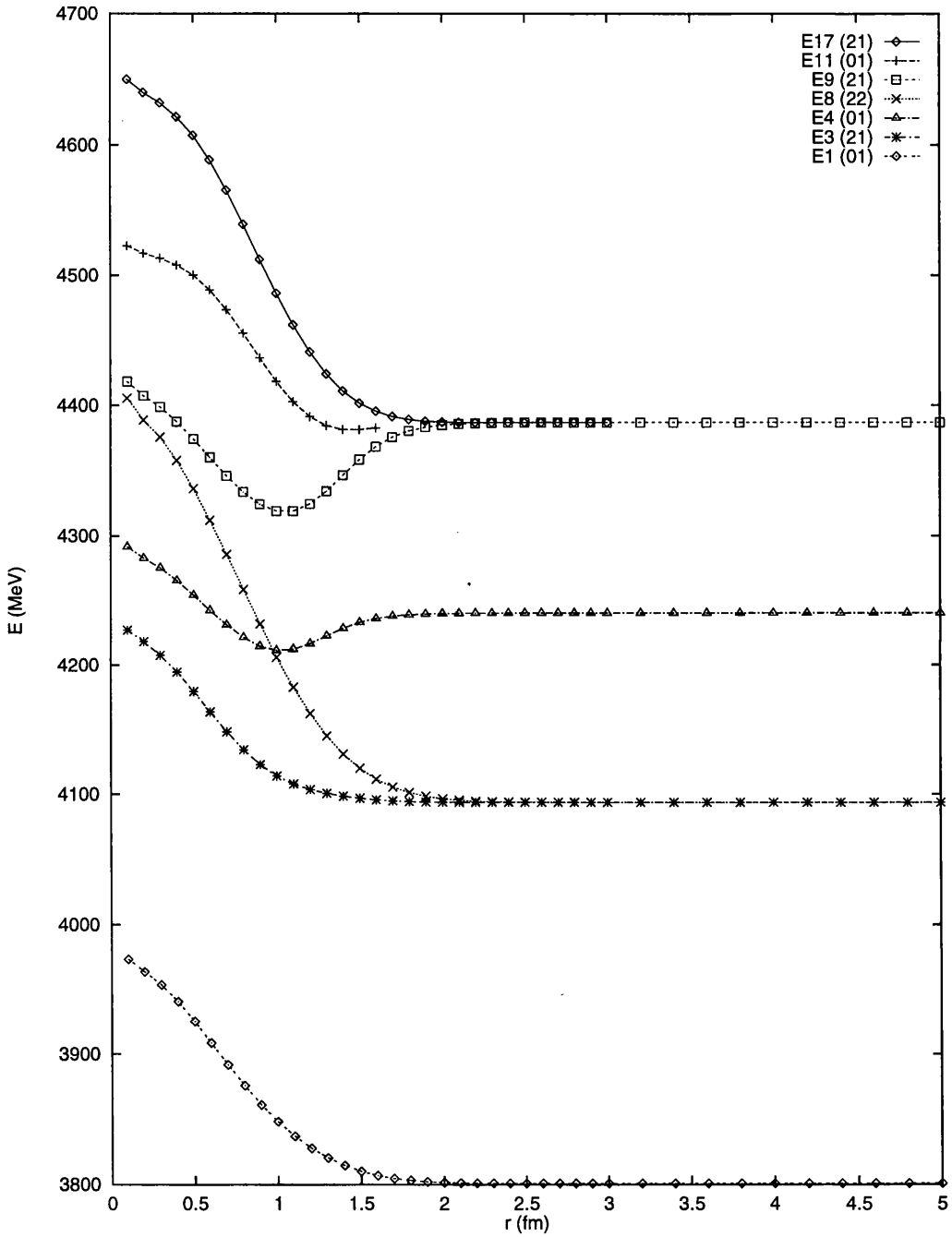

Figure 5.11: Spectrum with Oka-Yazaki Parameter set II. (even-$\boldsymbol{J}$ states)

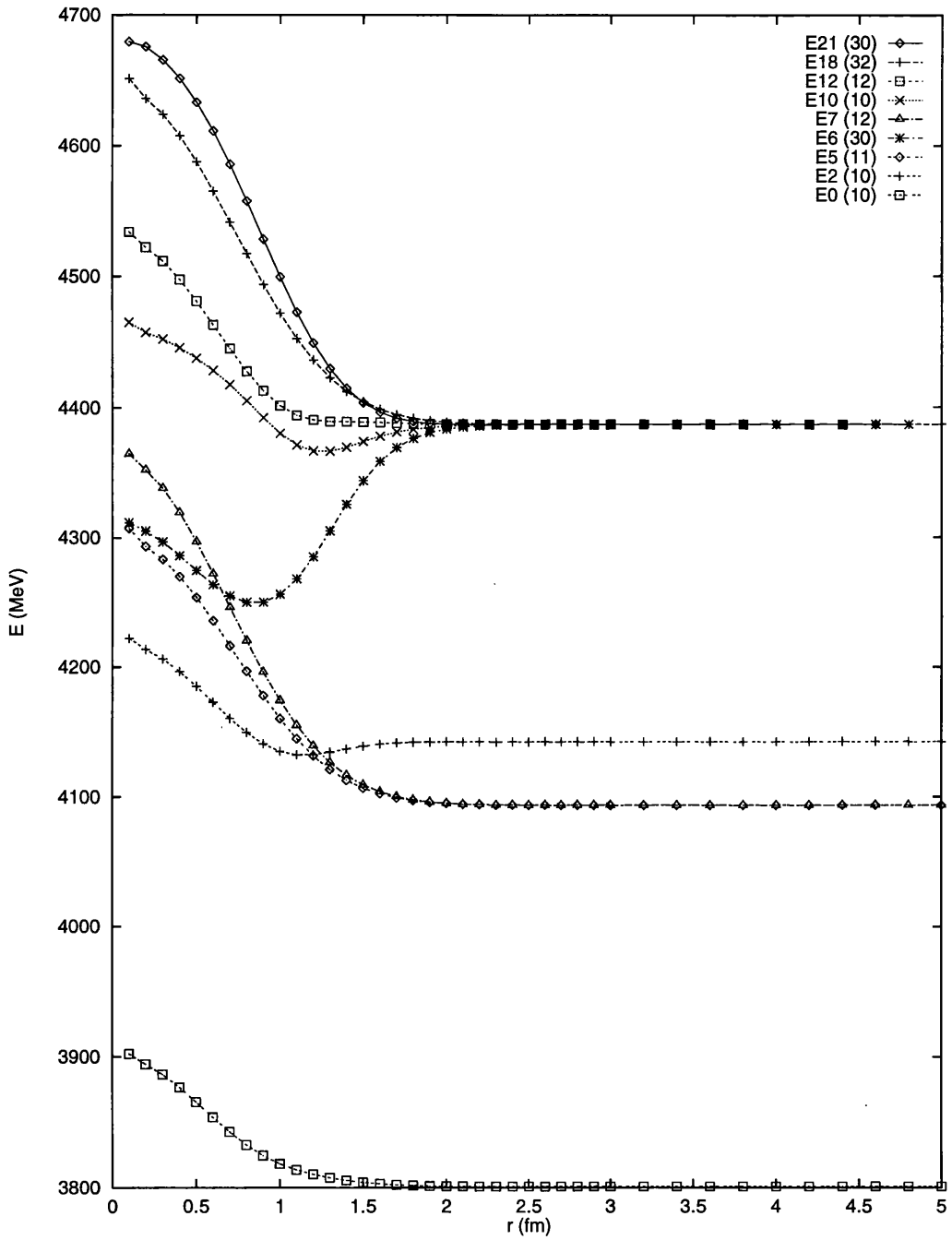

Figure 5.12: Spectrum with Oka-Yazaki Parameter set II. (odd-$\boldsymbol{J}$ states)

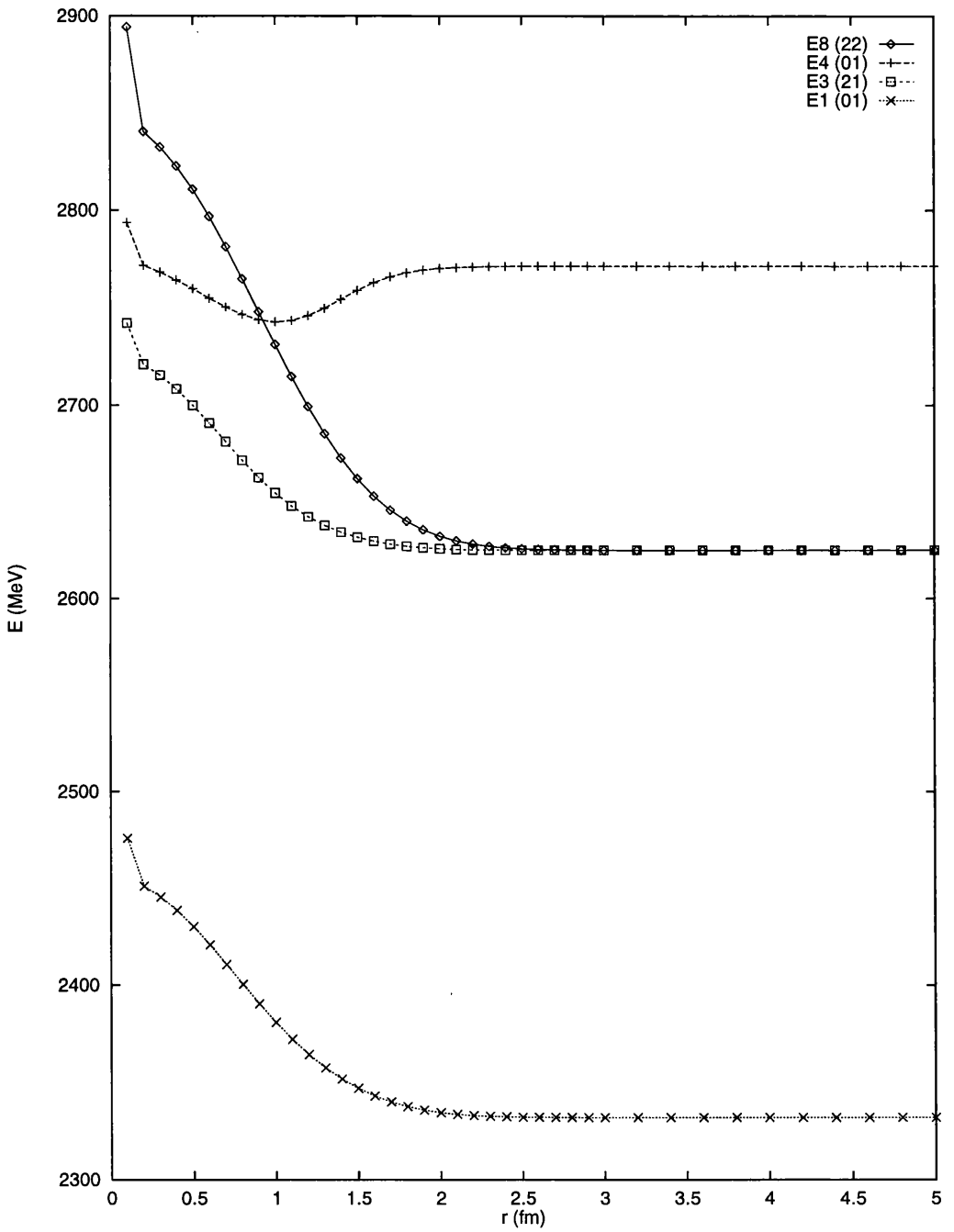

Figure 5.13: Spectrum with Oka-Yazaki Parameter set III. (even-$\boldsymbol{J}$ states)

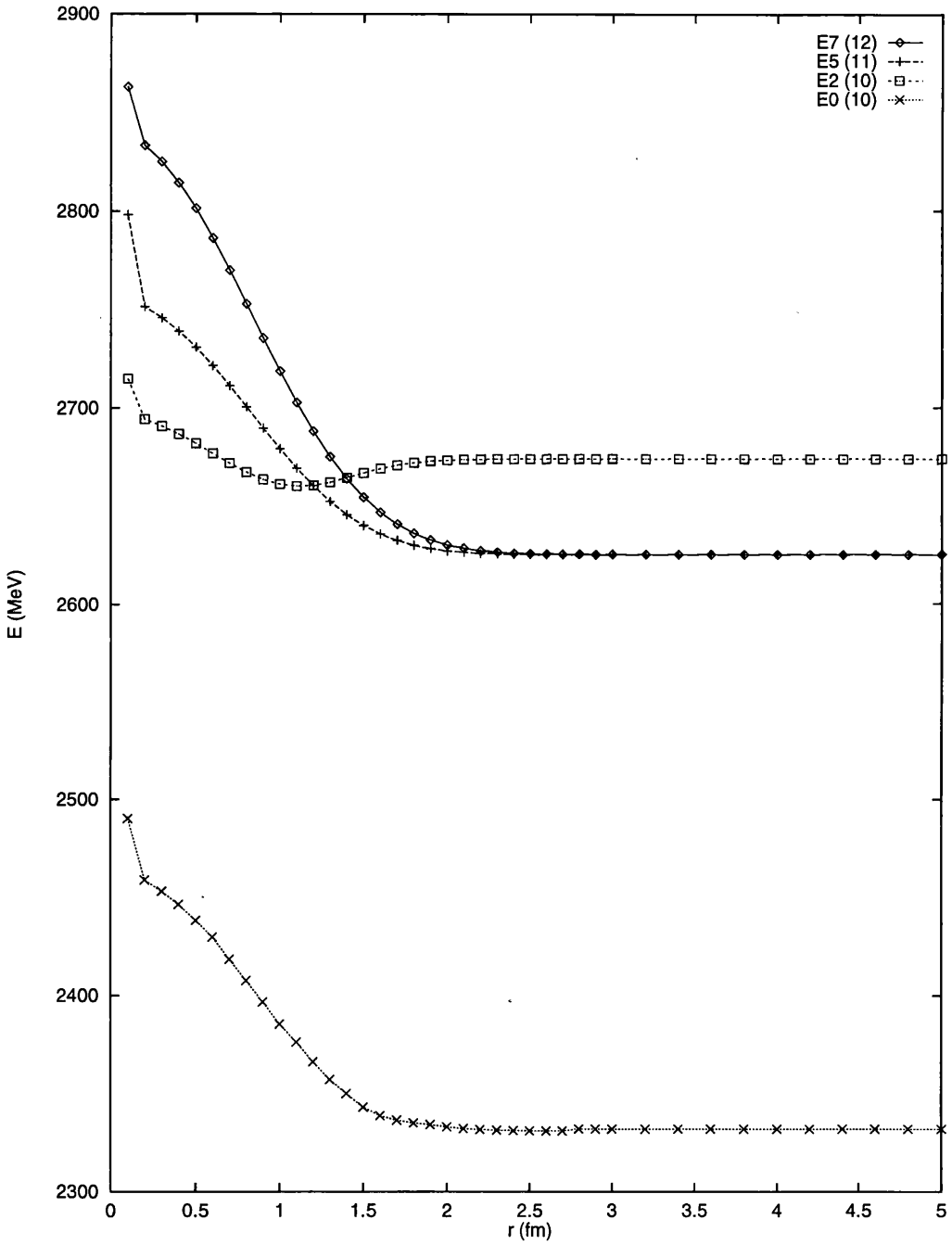

Figure 5.14: Spectrum with Oka-Yazaki Parameter set III. (odd-$\boldsymbol{J}$ states)

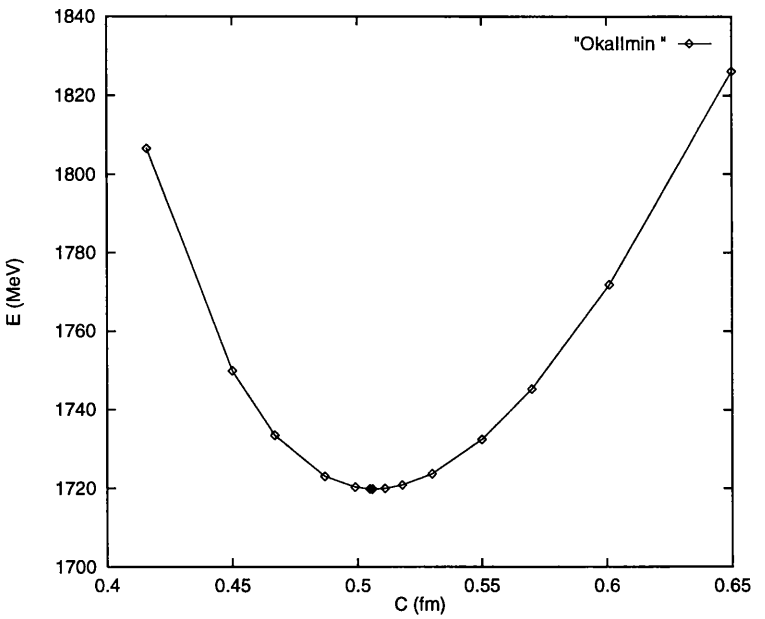

Figure 5.15: Nucleon mass minimisation (Parameter Set II)

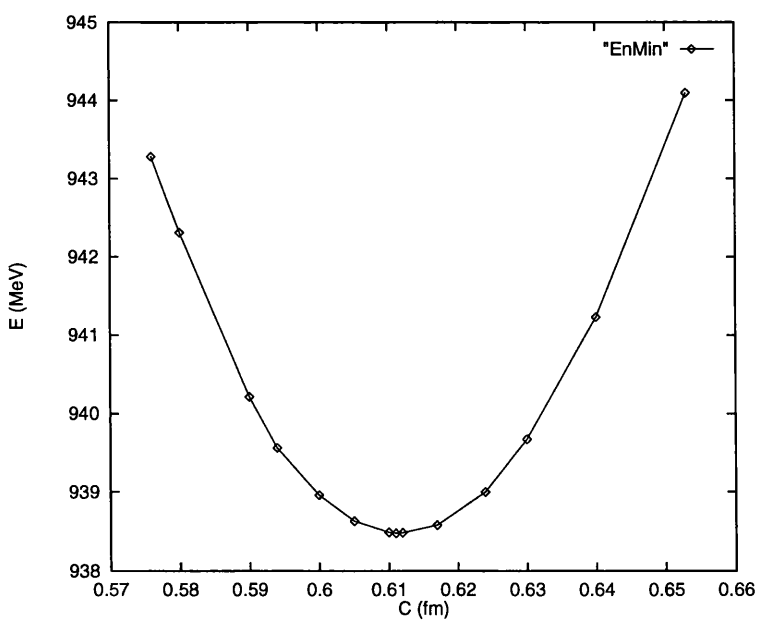

Figure 5.16: Nucleon mass minimisation with original parameters

	m_q MeV	α_s	a_c MeV/fm	C fm
I	313	0.878	234.0	0.5
II	313	1.517	135.3	0.6
III	313	2.409	85.2	0.7

Table 5.3: Oka-Yazaki quark model parameters

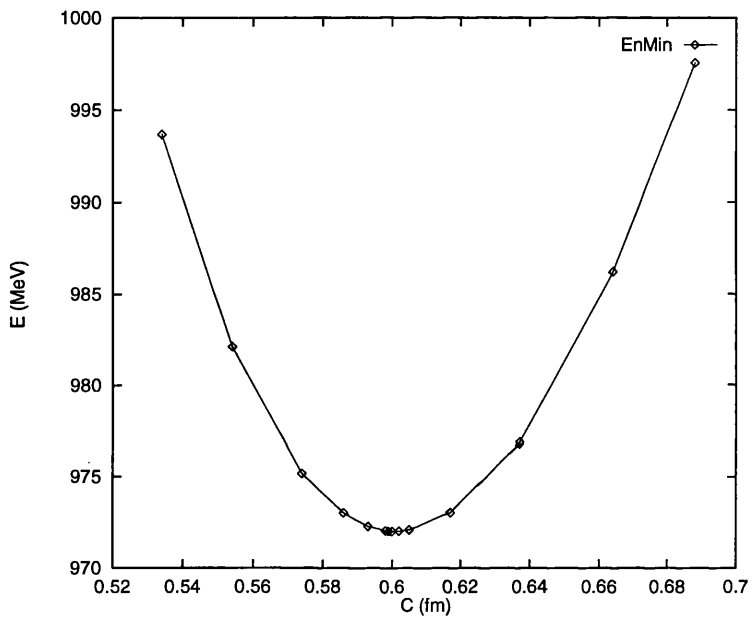

Figure 5.17: Nucleon mass minimisation with $a_c = 27.8\text{MeV/fm}^2$.

5.5 Quark-Antiquark Excitation Effects

We are now ready to move onto using the extended model space with $q\bar{q}$ excitations. The calculations are much larger and take considerably longer—the model space with $J_z = 0, T_z = 0$ has 8452 basis states, if we allow one meson to be created.

Another problem is that it is not immediately obvious in what way the parameters should be altered, if indeed they need to be at all. Hecht and Fujiwara have fitted a second set for calculations with $q\bar{q}$ pairs, shown in table 5.4, so we try both these, and our original set and compare the results. The spectra obtained are shown in figures 5.18 and 5.19.

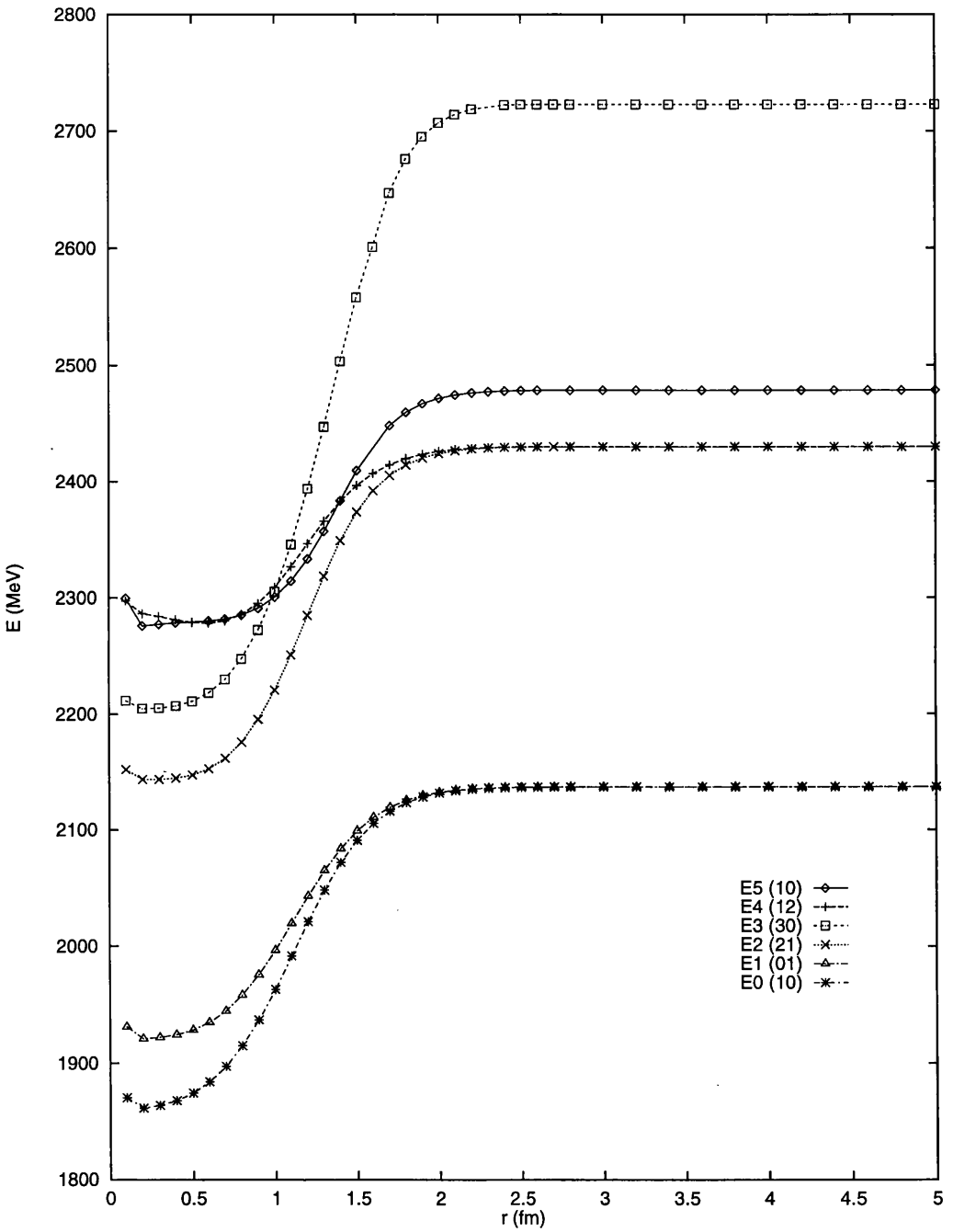

Figure 5.18: Extended space spectrum with original parameters (table 5.1)

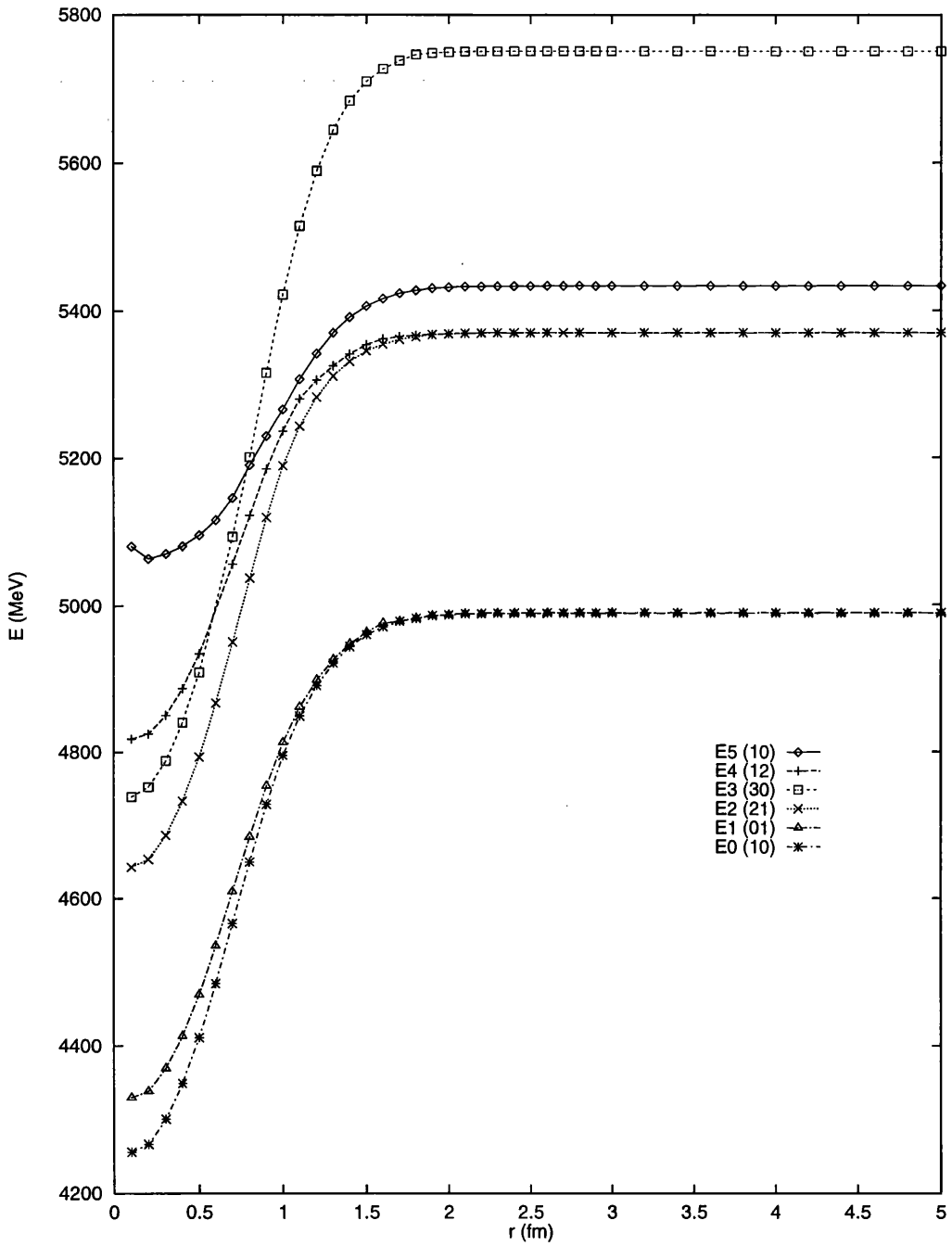

Figure 5.19: Extended space spectrum with parameters from table 5.4

m_q	α_s	a_c	C
471.2 MeV	2.973	335.7 MeV/fm^2	0.5235 fm

Table 5.4: Extended model parameters

It is possible to directly compare the states in figure 5.18 with those in figures 5.1 and 5.2, since both calculations used the same parameters, and immediately see the effects of the $q\bar{q}$ excitations. At large separations, where there is little overlap between the wavefunctions of each cluster, the $q\bar{q}$ effects become negligible and the states are the same as the six quark ones.

The most noticeable difference is that the energy at the origin has been lowered substantially (by about 200 MeV) in all the states. This effect was also observed by Hecht and Fujiwara, in [36], in comparing the potentials they obtained with the equivalent $6q$ potentials. These $6q$ calculations yielded much higher repulsive cores than ours, so the lowering of the energy left a small repulsive core of about 100 MeV.

In our case, the lowering of the energy has introduced an attraction at the origin, but the effect is broadly in agreement with their work. They record a lowering of about 600-800 MeV which, in fact, agrees very well with the results in figure 5.19 which we obtained using the same parameters, but why the core size in the $6q$ calculations should differ so much is not known. As before, the form of the interactions obtained is reasonably stable when the parameters are changed, although the depth and width of the potential wells vary quite a bit between sets.

5.5.1 Amplitudes of $6q(q\bar{q})$ in the Wavefunction

The graphs in figures 5.20 and 5.21 show the amplitude of $6q$ components in the eigenvectors obtained from the calculations we performed in the last section.

Examining these, we see that the proportion of the wavefunction containing $q\bar{q}$ varies similarly for all the states. It reaches a maximum of about 25% at small separations and tends to zero as the separation increases and the overlap between the wavefunctions becomes negligible. Obviously this happens sooner—at about 1.5 fm as opposed to 2.0 fm—for the system with the smaller

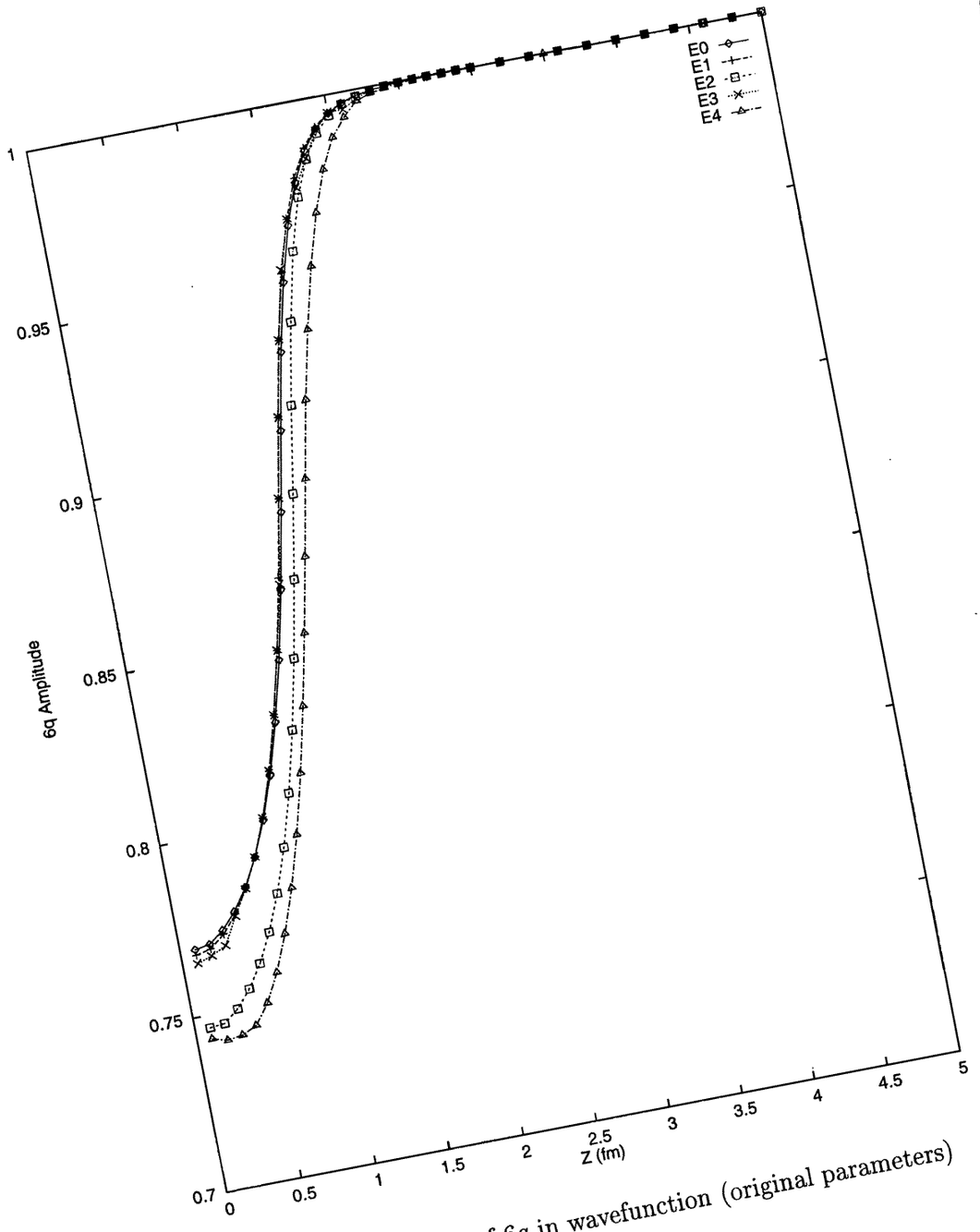

Figure 5.20: Proportion of 6q in wavefunction (original parameters)

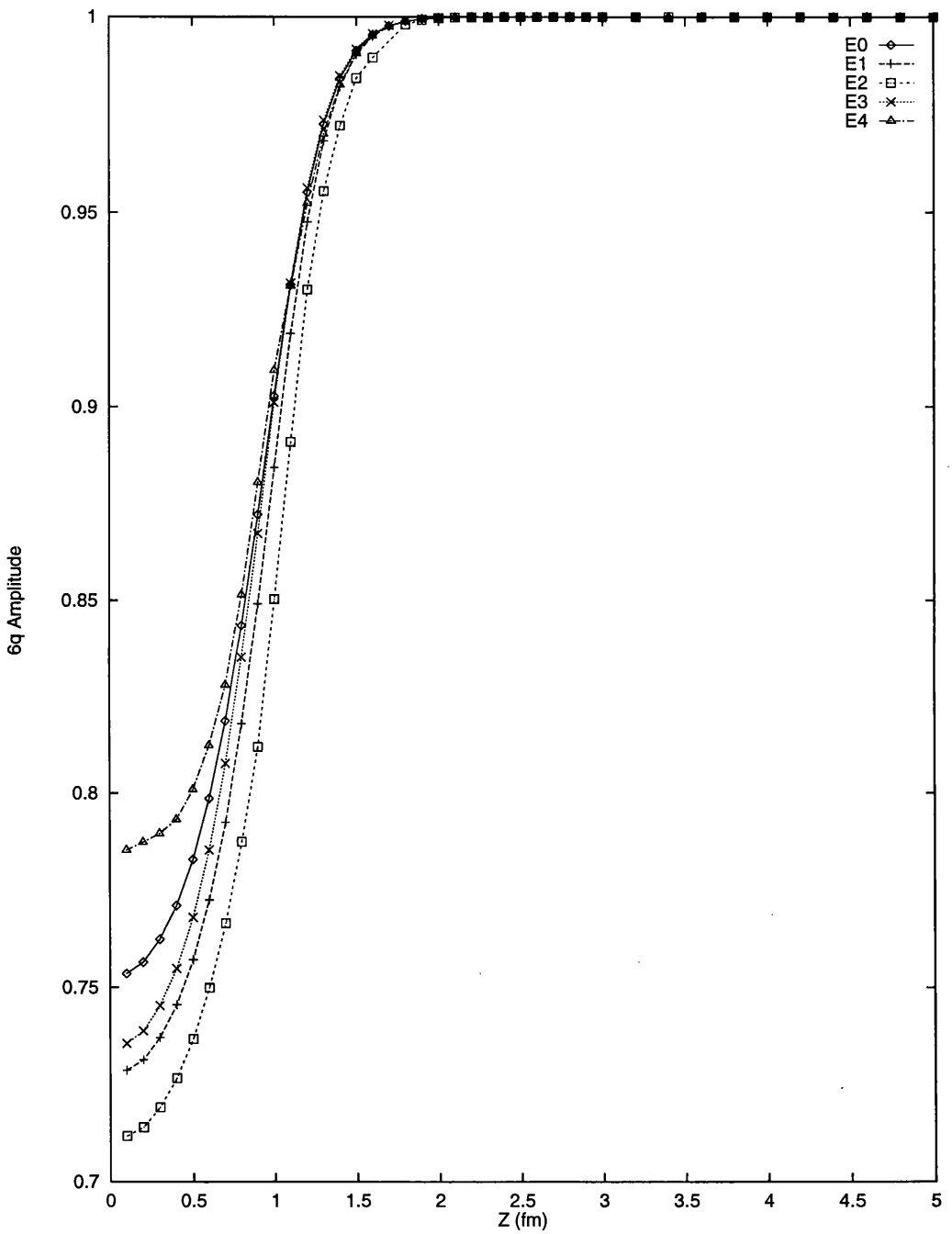

Figure 5.21: Proportion of $6q$ in wavefunction (parameters from table 5.4)

value of the oscillator parameter, which has narrower gaussian wavefunctions.

This doesn't mean that free nucleons do not have associated $q\bar{q}$ or pion clouds, but that the spatial wavefunctions of our model are too few to allow their excitation. Having 75% $6q$ components in the wavefunction near the origin is consistent with Ayat's results for the single nucleon [11] — she quotes a 73% $3q$ component in the nucleon wavefunction. We would not, however, expect the energy of our system at large separations to be consistent with her nucleon masses because of the difference in the spatial wavefunctions.

Chapter 6

Conclusion

6.1 Six Quark Calculations

Before attempting calculations in the full space, including $q\bar{q}$ pairs, we performed some studies along the same lines as the original $6q$ calculations done by Storm [28, 29]. The form of the new potential differs from the one she used but the results are qualitatively in agreement with those she obtained, displaying some degree of repulsion at the origin, with little or no attraction present. There does not seem to be any strong dependence in our results on the stability condition (section 5.4.3) which was imposed on the oscillator parameter in her work. This is a consequence of the fact that with our parameters, stability occurs at much higher (and more realistic) values ($C \approx 0.7\,\mathrm{fm}$) than Storm obtained with Harvey's system which gave $C \approx 0.2\,\mathrm{fm}$. The characteristic length is much longer and so quark distributions are more diffuse and changes occur more slowly with separation.

6.2 NN Interaction with $q\bar{q}$ Excitations

The main aim of the present work was to incorporate quark-antiquark excitations into the two-centre calculations and examine the resulting effects on the NN interaction. This was the next logical step in the series of quark model studies carried out using the Glasgow shell-model framework — combining the two-centre model with the cloudy baryon calculations performed

by Ayat [11, 38]. The main motivation is the possibility of obtaining results which are a first step towards describing the medium and long-range parts of the interaction with both the mesons and baryons modelled in terms of their constituent quarks.

Our calculations are currently restricted to S-states. The inclusion of excited states was possible in the single-centre calculations but is not currently feasible in the two-centre model because of the exponential increase in basis size which results. Ayat's calculations showed that the effects of such states on the baryon energies was substantially less than the changes brought about by including $q\bar{q}$ excitations. In addition, our aim is the study of the interaction between baryons, where the role of mesons is of prime importance, so hopefully such restrictions on the space should not be too serious.

As in Ayat's work, the inclusion of $q\bar{q}$ excitation effects was achieved by the use of an additional excitation potential (equation 4.3.1) which couples the $6q$ configurations to the new $6q(q\bar{q})$ configurations which have been included in the basis. Unlike Ayat we did not include strange quarks in the model, though most of the groundwork for this has been laid. The effects on the NN potential are unlikely to be of much significance and, in any case, the additional configurations would again lead to basis size problems.

In figures 6.1 and 6.2 we have plotted our various calculated NN potentials ($^1S_0$ and $^3S_1$ states) together with the corresponding Reid "soft-core" potentials [69], the formulae for which are reproduced in [70]. This is one of the most comprehensive fits available to the nucleon-nucleon potential and clearly displays the familiar features of this part of the interaction - it has an extremely strong repulsive region at short range, attraction at intermediate separations and takes on the form of the standard OPEP at long range. The $^3S_1$ diagram only includes the central term. There is an additional tensor component which leads to greater intermediate attraction in this channel but which has been omitted in our study. The tensor interaction could be included by adding additional terms from the qq or $q \rightarrow qq\bar{q}$ interactions but the additional complexity did not seem justified at this stage.

It has already been shown that the short-range repulsion of the interaction can be reproduced, at least to some extent, by studies using a pure $6q$ model. Most implementations have shown that a simple gluon-exchange po-

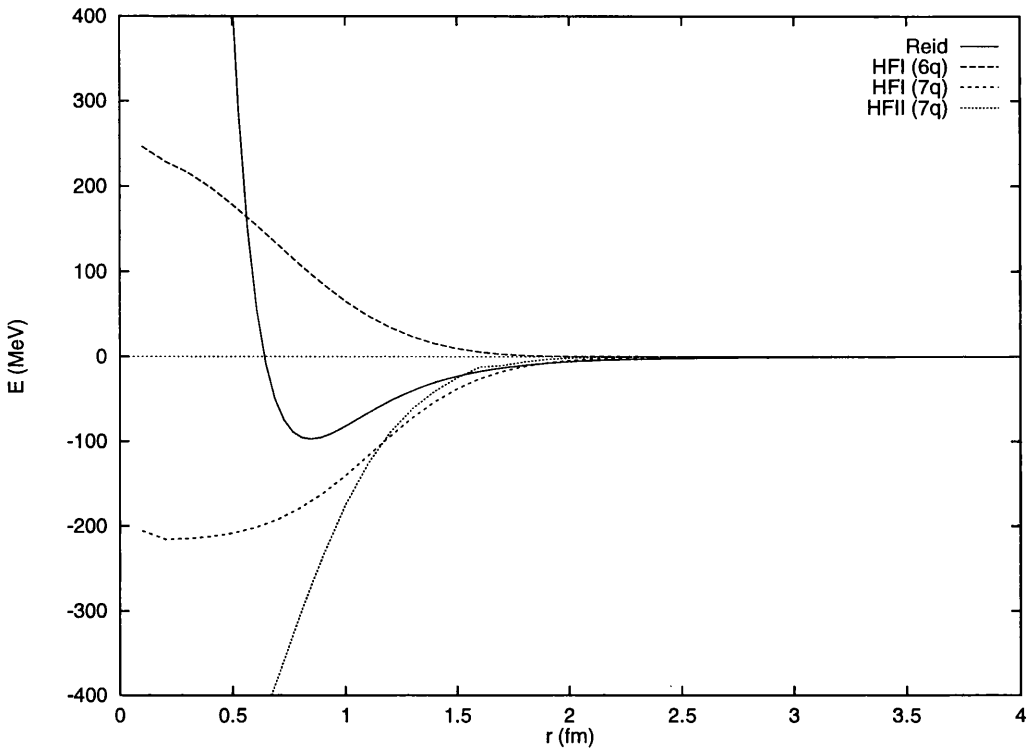

Figure 6.1: Comparison of $^1S_0$ Calculated States with Reid Potential

tential does lead to a repulsive core of limited height, even if it cannot be regarded as a complete description of the processes involved; meson effects are also important at short range — the exchange of the heavy ω vector meson is effective at distances less than 1 fm and is thought to provide a strong repulsive contribution.

It can be seen from figures 6.1 and 6.2 that the effect of including the $6q(q\bar{q})$ configurations is purely attractive. There is no longer any short-range repulsion. Instead both channels (with both sets of parameters) exhibit a strong negative potential at the origin, so the model in its present form does not reproduce any of the repulsive effects of meson exchange. Moving away from the origin, the potential becomes less attractive and decays to zero, as would be expected. The potentials all have a similar range which is consistent with the Reid potential. This is mainly governed by the oscillator parameter which defines the separations for which there will be an overlap between the two clusters. The lower oscillator parameter (*i.e.* $C = 0.5235\,\text{fm}$, from Hecht

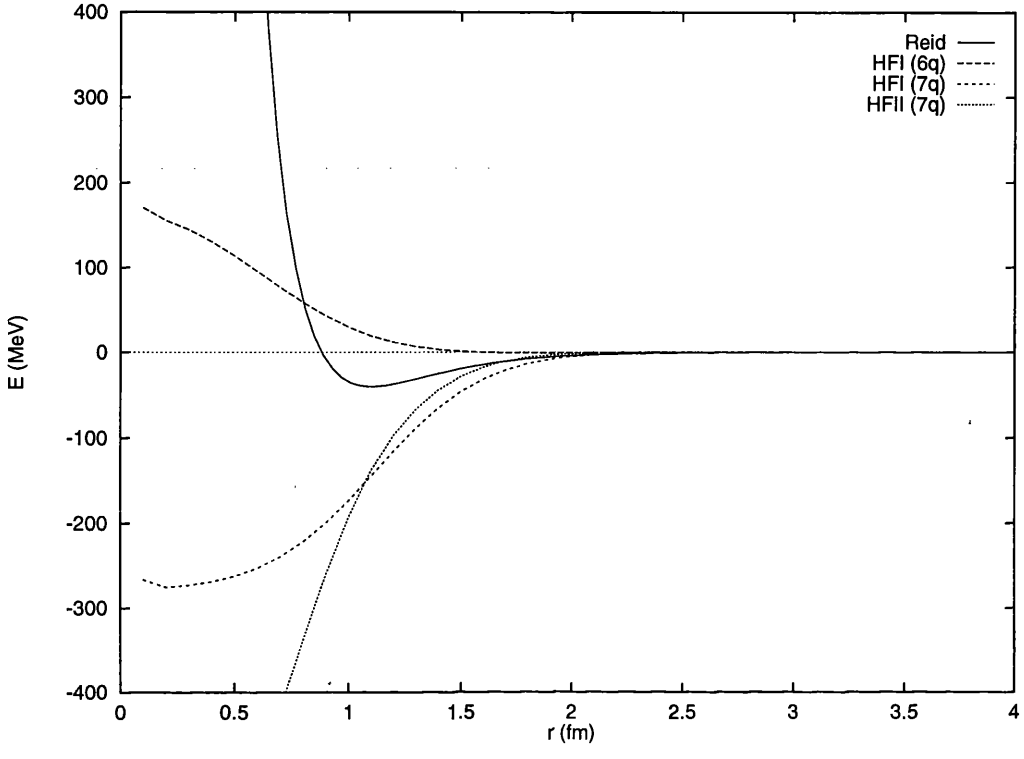

Figure 6.2: Comparison of $^3S_1$ Calculated States with Reid Potential

and Fujiwara's second set of parameters) gives the best fit of the two to the long-range part, though it does not match it exactly. The potentials are not very stable with changes in parameters — there is quite a large variation between the two sets. The second set displays much stronger attraction at the origin and decays more rapidly at intermediate distances.

For $6q$ calculations these results are comparable to other studies and reproduce the expected features of the interaction but the extension to the system with meson-excitations has not been very successful and may be pushing things too far.

In particular, the feasibility of describing mesons as simple $q\bar{q}$ pairs is questionable. A $q\bar{q}$ pair with the quantum numbers of a pion cannot be expected to give a realistic description of a real pion, with the correct pion mass, so a fit to the long-range OPEP is not likely to be readily achievable. This is likely to be exacerbated by the inadequacy of the spatial wavefunctions which limit the range of the potential and do not allow meson exchange to take place

when the clusters are separated. The nature of the region of overlap is likely to be of crucial importance when we are trying to model the interaction at intermediate and long range and the oscillator wavefunctions we use may not provide a sufficiently detailed description of this distinctive area. Our Gaussian wavefunctions fall off too rapidly with distance and simple exponentials would be better from the purely physical point of view. However, as explained earlier, it is very much easier ot calculate matrix elements with Gaussians and it was decided that for an exploratory investigation of this kind, Gaussians would be sufficient to give qualitative results.

At intermediate separations, the situation becomes too complicated for our model to handle. The heavier mesons come into play and two-pion exchange becomes important. Even if it were possible to include two meson configurations in the model, it is unlikely that simple $q\bar{q}$ excitations will provide an adequate description of the meson exchange process. The different lifetimes and masses of the mesons and resonances involved will all have an effect on the interaction which we cannot reproduce.

A more realistic model would also take account of the colour singlet nature of the exchanged meson – our model only specifies that the overall system should be a colour singlet but it should be possible for a colour singlet meson to be exchanged between two separated clusters and there is currently no mechanism in the model to enable this.

Although the shell-model approach allows a larger set of basis states to be used than for similar resonating group method calculations, the results are more difficult to analyse. In their RGM studies, Hecht and Fujiwara were eventually able to separate the contributions of different meson exchanges to the overall interaction, comparing the separate potentials with those of conventional meson theory [71]. This is not possible with our model where the results cannot readily be interpreted in this way. However, the form of the effective potentials which they obtained is broadly in agreement with our results, as can be seen from figure 2a of reference [71]. The main disagreement, as we have mentioned before, is in the amount of repulsion obtained in the pure $6q$ model where they have a repulsive core which is nearly 700 MeV at the origin, much larger than we obtain. Their $6q(q\bar{q})$ channel is purely attractive (-300 MeV at the origin) and this effect is increased with the addition of the

$6q(q\bar{q})^2$ channel which has a value of about -500 MeV at the origin. Initial attempts we have made at performing calculations with two $q\bar{q}$ excitations added to the model give results similar to these, showing largely increased attraction in the overall potential.

Another weakness of the model is the non-relativistic approximation, the validity of which is doubtful. As we mentioned earlier, it is surprising that the non-relativistic quark model produces such good results in $3q$ and $6q$ systems, but even if it is possible to hide relativistic effects in such cases, it doesn't necessarily follow that the approximation can be extended to include meson effects.

It was hoped that by performing calculations with $q\bar{q}$ excitations, it would be possible to expand on the limited information which the $6q$ model is capable of providing and to extend the non-relativistic quark model to study some of the effects of meson exchange on the NN interaction. However, the results we have obtained are not particularly enlightening and do not really provide any further insight into the nature of the interaction. The apparent failure of the model at this stage leads us to doubt its suitability for work of this kind and brings into question the wisdom of pursuing further studies in this area.

Appendix A

Spatial Matrix Element Integrals

This section contains details of the calculation of the matrix elements for the various spatial operators in the model. Recall that,

$$f_\mu = \frac{1}{(\pi C^2)^{3/4}} \exp\left(-\frac{(\boldsymbol{x} - \mu\frac{Z}{2}\boldsymbol{e}_3)^2}{2C^2}\right) \tag{A.0.1}$$

where $\mu = \pm 1$.

We perform the integrals in the basis f_μ and then use the transformation given in section 3.5.1 to change the matrix elements to the basis ϕ_μ which is used in the calculations.

A.1 Some Useful Results

We will make use of these results in what follows. The hyperbolic sine and cosine integrals can be found in [72].

$$\int_{-\infty}^{\infty} e^{-\frac{x^2}{c^2}}\, dx = c\sqrt{\pi} \tag{A.1.1}$$

$$\int_{-\infty}^{\infty} x e^{-cx^{\pm 2}}\, dx = 0 \tag{A.1.2}$$

$$\int_{-\infty}^{\infty} x^2 e^{x^2}\, dx = \frac{\sqrt{\pi}}{2} \tag{A.1.3}$$

$$\int_0^\infty e^{-\beta x^2} \sinh(\gamma x)\, dx = \frac{1}{2}\sqrt{\frac{\pi}{\beta}} \exp\frac{\gamma^2}{4\beta} \mathrm{erf}\left(\frac{\gamma}{2\sqrt{\beta}}\right) \tag{A.1.4}$$

$$\int_0^\infty x^2 e^{-\beta x^2} \sinh(\gamma x)\, dx = \sqrt{\pi}\frac{2\beta+\gamma^2}{8\beta^2\sqrt{\beta}} \exp\frac{\gamma^2}{4\beta} \mathrm{erf}\left(\frac{\gamma}{2\sqrt{\beta}}\right) + \frac{\gamma}{4\beta^2} \tag{A.1.5}$$

$$\int_0^\infty x e^{-\beta x^2} \sinh(\gamma x)\, dx = \frac{\gamma}{4\beta}\sqrt{\frac{\pi}{\beta}} \exp\frac{\gamma^2}{4\beta} \tag{A.1.6}$$

$$\int_0^\infty x^2 e^{-\beta x^2} \cosh(\gamma x)\, dx = \sqrt{\pi}\frac{2\beta+\gamma^2}{2\beta^2\sqrt{\beta}} \exp\frac{\gamma^2}{4\beta} \tag{A.1.7}$$

By differentiating A.1.6 with respect to β, we get

$$\int_0^\infty x^3 e^{-\beta x^2} \sinh(\gamma x)\, dx = \frac{\gamma}{4\beta}\sqrt{\frac{\pi}{\beta}} \exp\frac{\gamma^2}{4\beta} \left(\frac{3}{2\beta} + \frac{\gamma^2}{4\beta}\right) \tag{A.1.8}$$

Differentiating A.1.4 with respect to γ gives

$$\int_0^\infty x e^{\beta x^2} \cosh(\gamma x)\, dx = \sqrt{\frac{\pi}{\beta^3}}\frac{\gamma}{4} \exp\left(\frac{\gamma^2}{4\beta}\right) \mathrm{erf}\left(\frac{\gamma}{2\sqrt{\beta}}\right) + \frac{1}{2\beta} \tag{A.1.9}$$

The *error function*, $\mathrm{erf}(x)$, is defined by

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2}\, du \tag{A.1.10}$$

Where this occurs, it is approximated in the program by using the formula [73]

$$\mathrm{erf}(x) = \frac{1}{(1 + a_1 x + a_2 x^2 + \ldots + a_6)^{16}} + \epsilon(x) \tag{A.1.11}$$

Where

$$(\mathrm{mod}\ \epsilon(x)) \leq 3 \times 10^{-7} \tag{A.1.12}$$

$$\begin{array}{llll} a_1 = 0.0705230784 & a_2 = 0.0422820123 \\ a_3 = 0.0092705272 & a_4 = 0.0001520143 \\ a_5 = 0.0002765672 & a_6 = 0.0000430638 \end{array} \tag{A.1.13}$$

We have also used

$$\frac{d}{dx}\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \tag{A.1.14}$$

when differentiating the above integrals.

A.2 Miscellaneous Operators

A.2.1 The Operator '$\boldsymbol{x}$'

This operator is used in the calculation of the centre of mass operator matrix elements.

$$\begin{aligned}\langle f_\mu|\boldsymbol{x}|f_\nu\rangle &= \langle f_\mu|x_1\boldsymbol{e}_1 + x_2\boldsymbol{e}_2 + x_3\boldsymbol{e}_3|f_\nu\rangle \\ &= \frac{1}{(\pi C^2)^{\frac{3}{2}}}\int \boldsymbol{x} e^{-\frac{1}{2C^2}(2\boldsymbol{x}\cdot\boldsymbol{x}+\frac{1}{4}(\mu^2+\nu^2)Z^2-(\mu+\nu)Zx_3)}\,d^3\boldsymbol{x}\end{aligned}$$

The $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ components will be zero because the integrands are odd functions (see Equation A.1.2). The x_1 and x_2 integrals of the $\boldsymbol{e}_3$ component give a factor of πC^2 from Equation A.1.1. This leaves us with

$$\begin{aligned}\langle f_\mu|\boldsymbol{x}|f_\nu\rangle &= \sqrt{\pi}C\int_{-\infty}^{\infty} x_3 e^{-\frac{1}{2C^2}(2x_3^2-(\mu+\nu)Zx_3+\frac{Z^2}{2}}\,dx_3\,\boldsymbol{e}_3 \\ &= \sqrt{\pi}C\int_{-\infty}^{\infty} x_3 e^{-\frac{1}{C^2}(x_3-\frac{1}{4}Z(\mu+\nu))^2} e^{\frac{Z^2(\mu+\nu)^2}{16C^2}} e^{-\frac{Z^2}{4C^2}}\,dx_3\,\boldsymbol{e}_3 \\ &= \sqrt{\pi}Ce^{-\frac{Z^2}{4C^2}} e^{\frac{Z^2(\mu+\nu)^2}{16C^2}}\int_{-\infty}^{\infty}(y+\frac{(\mu+\nu)Z}{4})e^{-\frac{y^2}{C^2}}\,dy\,\boldsymbol{e}_3 \\ &= \frac{Z(\mu+\nu)}{4}e^{-\frac{Z^2(\mu-\nu)^2}{16C^2}}\boldsymbol{e}_3\end{aligned}$$

A.2.2 The Operator $\boldsymbol{p} = -i\hbar\boldsymbol{\nabla}$

The momentum operator occurs in both the kinetic energy and centre-of-mass operators.

$$\begin{aligned}\langle f_\mu|\boldsymbol{p}|f_\nu\rangle &= -\frac{i\hbar}{C^2}\langle f_\mu|x_1\boldsymbol{e}_1 + x_2\boldsymbol{e}_2 - (x_3-\frac{\nu Z}{2})\boldsymbol{e}_3|f_\nu\rangle \\ &= \frac{i\hbar}{(\sqrt{\pi}C)^3}e^{-\frac{Z^2(\mu^2+\nu^2)}{8C^2}}\int_{-\infty}^{\infty}(x_3-\frac{\nu Z}{2})e^{-\frac{1}{C^2}(x_3^2-\frac{Z(\mu+\nu)}{2}x_3)}\,dx_3\,\boldsymbol{e}_3 \\ &= \frac{i\hbar}{(\sqrt{\pi}C)^3}e^{-\frac{Z^2}{4C^2}}\int_{-\infty}^{\infty}(y+\frac{Z(\mu+\nu)}{4}-\frac{\nu Z}{2})e^{-\frac{y^2}{C^2}}e^{\frac{Z^2(\mu+\nu)^2}{16C^2}}\,dy\,\boldsymbol{e}_3 \\ &= \frac{i\hbar}{4C^2}Z(\mu-\nu)e^{-\frac{Z(\mu-\nu)^2}{16C^2}}\boldsymbol{e}_3\end{aligned}$$

A.2.3 The Operator x^2

This is part of the centre of mass operator.

$$\begin{aligned}
\langle f_\mu | \boldsymbol{x}^2 | f_\nu \rangle &= \frac{i\hbar}{(\sqrt{\pi}C)^3} e^{-\frac{Z^2}{4C^2}} \int (\boldsymbol{x}\cdot\boldsymbol{x}) e^{-\frac{1}{C^2}(x_1^2+x_2^2)} e^{-\frac{1}{C^2}(x_3^2 - \frac{(\mu+\nu)}{2} Z x_3)}\, d^3\boldsymbol{x} \\
&= \frac{i\hbar}{\sqrt{\pi}C} e^{-\frac{Z(\mu-\nu)^2}{16C^2}} \left(2\int_{-\infty}^{\infty} x^2 e^{\frac{x^2}{C^2}}\, dx \right. \\
&\quad \left. + \int_{-\infty}^{\infty} y^2 + \frac{(\mu+\nu)^2 Z^2}{16} e^{-\frac{y^2}{C^2}}\, dy \right) \\
&= e^{-\frac{Z(\mu-\nu)^2}{16C^2}} \left(\frac{3C^2}{2} + \frac{(\mu+\nu)^2}{16} Z^2 \right)
\end{aligned}$$

A.2.4 The Operator p^2

This one-body operator is needed to obtain matrix elements of kinetic energy. We have

$$\begin{aligned}
\langle f_\mu | \boldsymbol{p}^2 | f_\nu \rangle &= \langle f_\mu | \frac{3}{C^2} - \frac{\nu^2 Z^2}{4C^4} - \frac{\boldsymbol{x}^2}{C^4} + \frac{\nu Z x_3}{C^4} | f_\nu \rangle \\
&= \left[\frac{3}{C^2} - \frac{\nu^2 Z^2}{4C^4} - \frac{1}{C^4} \left(\frac{3C^2}{2} + \frac{(\mu+\nu)^2 Z^2}{16} \right. \right. \\
&\quad \left. \left. - \nu Z^2 \frac{(\mu+\nu)}{4} \right) \right] e^{-\frac{Z(\mu-\nu)^2}{16C^2}} \\
&= \left(\frac{3}{2C^2} - \frac{Z^2}{24C^2} (\mu-\nu)^2 \right) e^{-\frac{Z(\mu-\nu)^2}{16C^2}}
\end{aligned}$$

A.3 Integrals for the Main Two-Body Interaction

In this section, we deal with the operators in the qq interaction potental. The integrals are all of the form

$$\begin{aligned}
&\langle f_{\mu_1} f_{\mu_2} \mid V(\boldsymbol{r}) \mid f_{\nu_1} f_{\nu_2} \rangle = \\
&\quad \frac{1}{\pi^3 C^6} \int V(|\boldsymbol{x}_1 - \boldsymbol{x}_2|) e^{-\frac{1}{2C^2}(\boldsymbol{x}_1 - \frac{\mu_1 Z}{2}\boldsymbol{e}_3)^2} \\
&\quad e^{-\frac{1}{2C^2}(\boldsymbol{x}_2 - \frac{\mu_2 Z}{2}\boldsymbol{e}_3)^2} e^{-\frac{1}{2C^2}(\boldsymbol{x}_1 - \frac{\nu_1 Z}{2}\boldsymbol{e}_3)^2} e^{-\frac{1}{2C^2}(\boldsymbol{x}_2 - \frac{\nu_2 Z}{2}\boldsymbol{e}_3)^2}\, d^3\boldsymbol{x}_1 d^3\boldsymbol{x}_2 \quad \text{(A.3.1)}
\end{aligned}$$

In obvious notation, the argument of the exponential in this integral is a factor of $-1/2C^2$ times

$$A_{exp} = (\boldsymbol{x}_1 - a_1\boldsymbol{e}_3)^2 + (\boldsymbol{x}_2 - a_2\boldsymbol{e}_3)^2 + (\boldsymbol{x}_1 - b_1\boldsymbol{e}_3)^2 + (\boldsymbol{x}_2 - b_2\boldsymbol{e}_3)^2 \quad \text{(A.3.2)}$$

In situations like this, where two particles are interacting via an internal force, the centre of mass will move like a free particle and the relative motion will be determined by the interaction potential, $V(\boldsymbol{r})$. The standard approach is to transform to relative and centre of mass coordinates:

$$\boldsymbol{R} = \frac{1}{\sqrt{2}}(\boldsymbol{x}_1 + \boldsymbol{x}_2) \quad \text{(A.3.3)}$$

$$\boldsymbol{\rho} = \frac{1}{\sqrt{2}}(\boldsymbol{x}_1 - \boldsymbol{x}_2) \quad \text{(A.3.4)}$$

This becomes

$$\begin{aligned} A_{exp} &= 2(\boldsymbol{\rho}^2 + \boldsymbol{R}^2) - \sqrt{2}(a_1 + a_2 + b_1 + b_2)\boldsymbol{R}\cdot\boldsymbol{e}_3 \\ &\quad -\sqrt{2}(a_1 + b_1 - a_2 - b_2)\boldsymbol{\rho}\cdot\boldsymbol{e}_3 + Z^2 \\ &= 2(\boldsymbol{R} - \frac{a_1 + a_2 + b_1 + b_2}{2\sqrt{2}}\boldsymbol{e}_3)^2 \\ &\quad +2(\boldsymbol{\rho}^2 - \frac{a_1 + b_1 - a_2 - b_2}{\sqrt{2}}\boldsymbol{\rho}\cdot\boldsymbol{e_3}) + Z^2 \end{aligned}$$

The integral with respect to $\boldsymbol{R}$ reduces to

$$\int_0^\infty R^2 e^{\frac{R^2}{C^2}}\, dR \int_0^\pi \sin(\theta)\, d\theta \int_0^{2\pi} d\phi = (\pi C^2)^{3/2} \quad \text{(A.3.5)}$$

The integral with respect to $\boldsymbol{\rho}$ is given by

$$2\pi \int_0^\infty \int_0^\pi \rho^2 V(\sqrt{2}\rho) e^{\frac{\rho^2}{C^2}} e^{\frac{1}{\sqrt{2}C^2}(a_1+b_1-a_2-b_2)\rho\cos\theta} \rho^2 \sin\theta\, d\rho\, d\theta \quad \text{(A.3.6)}$$

If we simplify this by changing variables from θ to $\cos\theta$, the full integral, A.3.1, can now be written as

$$\begin{aligned} &\langle f_{\mu_1} f_{\mu_2} \mid V(\boldsymbol{r}) \mid f_{\nu_1} f_{\nu_2} \rangle = \\ &\quad e^{-\frac{Z^2}{2C^2}\left(1 - \frac{(\mu_1+\mu_2+\nu_1+\nu_2)^2}{16}\right)} \underbrace{\frac{4}{\sqrt{\pi}c^2\gamma} \int_0^\infty e^{-\frac{\rho^2}{C^2}} \rho V(\sqrt{2}\rho) \sinh(\gamma\rho)\, d\rho}_{I} \end{aligned} \quad \text{(A.3.7)}$$

where

$$\gamma = \frac{Z}{2\sqrt{2}C^2}(\mu_1 + \nu_1 - \mu_2 - \nu_2) \quad \text{(A.3.8)}$$

Now all we have to do is work out the integral I, indicated in A.3.7, for the terms, $V(r)$, in the potential.

A.3.1 $V(r) = 1/r$

Here we have

$$I = \frac{4}{\sqrt{\pi}c^3\gamma\sqrt{2}}\int_0^\infty e^{-\frac{\rho^2}{C^2}}\sinh(\gamma\rho)\,d\rho \tag{A.3.9}$$

$$= \frac{\sqrt{2}}{\gamma C^2}e^{\frac{\gamma^2C^2}{4}}\mathrm{erf}\left(\frac{\gamma C}{2}\right) \tag{A.3.10}$$

by using A.1.4

A.3.2 $V(r) = r$

$$I = \frac{4\sqrt{2}}{\sqrt{\pi}c^3\gamma}\int_0^\infty \rho^2 e^{-\frac{\rho^2}{C^2}}\sinh(\gamma\rho)\,d\rho \tag{A.3.11}$$

$$= \frac{1}{\sqrt{2}\gamma}\left(2+\gamma^2C^2\right)e^{\frac{\gamma^2C^2}{4}}\mathrm{erf}\left(\frac{\gamma C}{2}\right)+\sqrt{\frac{2}{\pi}}C \tag{A.3.12}$$

A.3.3 $V(r) = r^2$

$$I = \frac{8}{\sqrt{\pi}c^3\gamma\sqrt{2}}\int_0^\infty \rho^3 e^{-\frac{\rho^2}{C^2}}\sinh(\gamma\rho)\,d\rho \tag{A.3.13}$$

$$= 2C^2\left(\frac{3}{2}+\frac{\gamma^2c^2}{4}\right)e^{\frac{\gamma^2C^2}{4}} \tag{A.3.14}$$

A.3.4 $V(r) = \delta^3(\boldsymbol{r})$

The three dimensional delta function is equivalent to

$$\delta^3(\boldsymbol{x}) = \delta(x_1)\delta(x_2)\delta(x_3) \tag{A.3.15}$$

With

$$\int \delta^3(\boldsymbol{x})d^3(\boldsymbol{x}) = 1 \tag{A.3.16}$$

In polar coordinates

$$\delta^3(\boldsymbol{r}) = \frac{1}{4\pi r^2}\delta(r)$$

$$= \frac{1}{8\pi\rho^2}\delta(\sqrt{2}\rho)$$

$$= \frac{1}{8\sqrt{2}\pi\rho^2}\delta(\rho) \tag{A.3.17}$$

Since

$$\delta(ax) = \frac{1}{a}\delta(x) \tag{A.3.18}$$

So

$$\begin{aligned} I &= \frac{1}{(2\pi)^{3/2}C^3\gamma}\int_0^\infty e^{\frac{\rho^2}{C^2}}\frac{\sinh(\gamma\rho)}{\rho}\delta(\rho)d\rho \\ &= \frac{1}{(2\pi C^2)^{\frac{3}{2}}} \end{aligned} \tag{A.3.19}$$

A.4 Integrals for the Yu-Zhang Potential

In this section, the integrals necessary for the use of the $q\bar{q}$ excitation potential, equation 4.3.1, are evaluated. As above, we change to relative and centre of mass coordinates to evaluate $\langle\, f_{\mu_1} f_{\mu_2} \mid V(\boldsymbol{r}) \mid f_{\nu_1} f_{\nu_2} \,\rangle$. The integral then splits into a term in $\boldsymbol{R}$

$$\frac{1}{(\sqrt{\pi}C^2)^{3/2}}e^{-\frac{(A-B)^2}{4C^2}}\int e^{\frac{1}{C^2}\left(\boldsymbol{R}-\frac{A+B}{2}\right)^2}v(\boldsymbol{R})\,d^3\boldsymbol{R} \tag{A.4.1}$$

multiplied by a term in $\boldsymbol{\rho}$

$$\frac{1}{(\sqrt{\pi}C^2)^{3/2}}e^{-\frac{(a-b)^2}{4C^2}}\int e^{\frac{1}{C^2}\left(\boldsymbol{\rho}-\frac{a+b}{2}\right)^2}V(\boldsymbol{\rho})\,d^3\boldsymbol{\rho} \tag{A.4.2}$$

where

$$A = \frac{1}{2\sqrt{2}}Z(\mu_1+\mu_2)\boldsymbol{e}_3 \tag{A.4.3}$$

$$a = \frac{1}{2\sqrt{2}}Z(\mu_1-\mu_2)\boldsymbol{e}_3 \tag{A.4.4}$$

$$B = \frac{1}{2\sqrt{2}}Z(\nu_1+\nu_2)\boldsymbol{e}_3 \tag{A.4.5}$$

$$b = \frac{1}{2\sqrt{2}}Z(\nu_1-\nu_2)\boldsymbol{e}_3 \tag{A.4.6}$$

The integrals in $\boldsymbol{R}$ which we have to deal with are trivial to evaluate. If we then change to polar coordinates, we can rewrite the integral in $\boldsymbol{\rho}$ as

$$\frac{2\pi}{(\pi C^2)^{3/2}}e^{\frac{1}{2C^2}(a^2+b^2)}\underbrace{\int e^{-\frac{\rho^2}{C^2}}e^{\frac{\rho(a+b)}{C^2}\cos\theta}V(\rho,\theta)\rho^2\sin\theta\,d\rho d\theta}_{I} \tag{A.4.7}$$

We now proceed to work out expressions for these integrals for each of the terms in the potential.

A.4.1 The Term $\boldsymbol{r}/r^3$

We can write this as $V(\rho,\theta) = \cos\theta/\rho^2$. The integral, I, from the previous section then becomes

$$I = \int_0^\infty e^{\frac{\rho^2}{C^2}}\, d\rho \int_0^\pi e^{\gamma\rho\cos\theta} \cos\theta \sin\theta\, d\theta \tag{A.4.8}$$

where $\gamma = (a+b)/C^2$ (the same as equation A.3.8. After integrating with respect to θ we get

$$\begin{aligned} I &= \int_0^\infty e^{\frac{\rho^2}{C^2}} \left[\frac{2}{\gamma\rho}\cosh(\gamma\rho) - \frac{sinh(\gamma\rho)}{\gamma\rho}\right] d\rho \\ &= \frac{2}{\gamma}\int_0^\infty e^{-\frac{u^2}{\gamma^2C^2}} \frac{2}{u}\frac{d}{du}\left(\frac{\sinh u}{u}\right) du \end{aligned}$$

Integrating by parts gives

$$I = -\frac{2}{\gamma} + \frac{4}{\gamma^3C^2}\int_0^\infty e^{\frac{u^2}{\gamma^2C^2}} \sinh u\, du \tag{A.4.9}$$

which can be solved using equation A.1.4 to give

$$I = -\frac{2}{\gamma} + \frac{2\sqrt{\pi}}{\gamma^2 C} e^{\frac{\gamma^2C^2}{4}} \operatorname{erf}\left(\frac{\gamma c}{2}\right) \tag{A.4.10}$$

With $v(\boldsymbol{R}) = 1$ in A.4.1, the integral in $\boldsymbol{R}$ just gives a factor

$$e^{-(A-B)^2/4C^2} \tag{A.4.11}$$

So the complete integral is then

$$\mathcal{I}_1 = \frac{1}{\gamma c^3} e^{-\frac{1}{2C^2}\left(\frac{(A-B)^2}{2}+a^2+b^2\right)} \left[e^{\frac{\gamma^2C^2}{4}} \frac{\operatorname{erf}\frac{\gamma C}{2}}{\frac{\gamma C}{2}} - \frac{2}{\sqrt{\pi}}\right] \tag{A.4.12}$$

A.4.2 The Term $\frac{\nabla_1}{r}$

This momentum operator only acts on particle 1 in the two particle wavefunction $|\, f_{\nu_1} f_{\nu_2}\,\rangle$. We have

$$\nabla_1 e^{-\frac{1}{2C^2}(\boldsymbol{x}_1 - \frac{\nu_1 Z}{2}\boldsymbol{e}_3)^2} = -\frac{1}{2C^2}\left(\boldsymbol{x}_1 - \frac{\nu_1 Z}{2}\boldsymbol{e}_3\right) e^{-\frac{1}{2C^2}(\boldsymbol{x}_1 - \frac{\nu_1 Z}{2}\boldsymbol{e}_3)^2} \tag{A.4.13}$$

So we can write

$$\frac{\nabla_1}{r} = -\frac{1}{2\rho C^2}\left(\boldsymbol{R} + \boldsymbol{\rho} - \frac{\nu_1 Z}{2}\boldsymbol{e}_3\right) \tag{A.4.14}$$

and we now have to work out the integral for each of the terms on the right of this equation.

A.4.3 The Term $\boldsymbol{R}/\rho$

Putting $v(\boldsymbol{R}) = \boldsymbol{R}$ in A.4.1, we find that the integral with respect to $\boldsymbol{R}$ is equal to

$$e^{-\frac{(A-B)^2}{4C^2}} \frac{1}{2} (\boldsymbol{A} + \boldsymbol{B}) \boldsymbol{e}_3 \tag{A.4.15}$$

The integral I from A.4.7 is

$$I = \int e^{-\frac{\rho^2}{C^2}} e^{\frac{\rho(a+b)}{C^2}\cos\theta} \rho \sin\theta \, d\rho d\theta \tag{A.4.16}$$

Integrating with respect to $\cos\theta$ leaves

$$I = \frac{2}{\gamma} \int_0^\infty e^{-\frac{\rho^2}{C^2}} \sinh \rho\gamma \, d\rho \tag{A.4.17}$$

with $\gamma = (a+b)/C^2$ as before. Using A.1.4 gives

$$I = \frac{\sqrt{\pi} C^2}{2} e^{\frac{\gamma^2 C^2}{4}} \frac{\operatorname{erf}(\frac{\gamma C}{2})}{\gamma C/2} \tag{A.4.18}$$

Combining the above results gives the result for the term $-\frac{1}{2C^2}\frac{\boldsymbol{R}}{\rho}$

$$-\frac{A+B}{4C^3} e^{-\frac{1}{2C^2}\left(\frac{(A-B)^2}{2}+a^2+b^2\right)} e^{\frac{\gamma^2 C^2}{4}} \frac{\operatorname{erf}\frac{\gamma C}{2}}{\frac{\gamma C}{2}} \tag{A.4.19}$$

A.4.4 The Term $\boldsymbol{\rho}/\rho$

The $\boldsymbol{R}$ integral is the same as A.4.11. We are then left with

$$I = \int e^{-\frac{\rho^2}{C^2}} e^{\frac{\rho(a+b)}{C^2}\cos\theta} \rho^2 \sin\theta \, d\rho d\theta \tag{A.4.20}$$

The θ integral is the same as in section A.4.1 leaving

$$I = \int_0^\infty e^{-\frac{\rho^2}{C^2}} \frac{2}{\gamma\rho} \left[\cosh(\gamma\rho) - \frac{\sinh(\gamma\rho)}{\gamma\rho}\right] \rho^2 \, d\rho \tag{A.4.21}$$

Using A.1.4 and A.1.9 and simplifying gives

$$I = \frac{1}{\gamma^3} \left[\sqrt{\pi} \frac{\gamma^2 C^2}{2} \left(\frac{\gamma^2 C^2}{2} - 1\right) e^{\frac{\gamma^2 C^2}{4}} \frac{\operatorname{erf}\frac{\gamma C}{2}}{\frac{\gamma C}{2}} + \gamma^2 C^2\right] \tag{A.4.22}$$

A.4.5 Term $\nu_1 Z e_3/(2^{1/2}\rho)$

Here the ρ integral takes exactly the same form as A.4.19 and the $\boldsymbol{R}$ integral is the same as A.4.11.

A.4.6 Summary

Gathering together the information from this section we see that the quantities we have to evaluate can be summarised as follows

$$I_1 = \frac{2}{\gamma C^3} T_{\text{expl}} \left[T_{\text{exp2}} T_{\text{erf}} - \frac{1}{\sqrt{\pi}} \right] \tag{A.4.23}$$

$$I_2 = -\frac{1}{2C^3} (A+B) T_{\text{expl}} T_{\text{exp2}} T_{\text{erf}} \tag{A.4.24}$$

$$I_3 = -\frac{1}{\gamma C^3} T_{\text{expl}} \left[\left(\frac{\gamma^2 C^2}{2} - 1 \right) T_{\text{exp2}} T_{\text{erf}} + \frac{1}{\sqrt{\pi}} \right] \tag{A.4.25}$$

$$I_4 = \frac{\nu_1 Z}{\sqrt{2} C^3} T_{\text{expl}} T_{\text{exp2}} T_{\text{erf}} \tag{A.4.26}$$

where

$$T_{\text{expl}} = e^{\frac{1}{2C^2}\left[\frac{(A-B)^2}{2} + a^2 + b^2\right]} \tag{A.4.27}$$

$$T_{\text{exp2}} = e^{\frac{\gamma^2 C^2}{4}} \tag{A.4.28}$$

$$T_{\text{erf}} = \frac{1}{\gamma C} \text{erf}\left(\frac{\gamma C}{2}\right) \tag{A.4.29}$$

These are the expressions which worked out in the computer program.

Appendix B

Binding Energy of the Deuteron

To illustrate the excessive binding resulting from the attraction evident in figures 5.18 and 5.19, we have calculated the binding energies of the deuteron ($^3S_1$) and the other neutron-proton state ($^1S_0$).

The Woods-Saxon form is used for the potential as it has roughly the same shape as the graphs in question

$$v(r) = -\frac{v_0}{1 + \exp\left[\frac{r-r_0}{a}\right]} \tag{B.0.1}$$

Here v_0 is the difference between the energy at large separation and at the minimum at $r \approx 0$. r_0 is the value of r at which $v(r) = -v_0/2$ and a is obtained from the gradient at $r = r_0$ using

$$\left.\frac{dv(r)}{dr}\right|_{r=r_0} = -\frac{v_0}{4a} \tag{B.0.2}$$

In figure 5.18, the graph E0 is approximated by $v_0 = 280$, $r_0 = 1.25$, $a = 0.25$. Inserting this potential in Schrödinger's equation for the relative wavefunction of a proton and a neutron with angular momentum $l = 0$

$$\frac{-\hbar^2}{2\mu}\frac{1}{r}\frac{d^2}{dr^2}(r\psi) + v(r)\psi = E\psi \tag{B.0.3}$$

gives $E = -110$MeV for the ground state, which does not compare well with -2.225MeV, the energy of the deuteron. In addition, there is a second state with $l = 1$ and energy -4.5MeV which does not exist in nature.

The graph E1 ($^1S_0$) is approximated by $v_0 = 215$, $r_0 = 1.25$ and $a = 0.25$. This time the state of lowest energy has $E = -66MeV$.

The parameter set of table 5.4 does not improve matters. For E0, we have $v_0 = 600$, $r_0 = 0.8$ and $a = 0.22$. Thus the attraction is much stronger though of shorter range. There is one bound state of energy -196MeV.

For E1, the potential is fitted by $v_0 = 530$, $r_0 = 0.8$ and $a = 0.22$, and the energy of the single bound state is -156MeV.

Appendix C

Technical Information

Here we give details of the program used to implement the model. The calculation can be roughly separated into the following stages:

- Constructing the single particle basis.
- Assembling the many-particle basis states.
- Finding the sets of orbit pairs for two-body transitions.
- Calculating the two-body matrix elements of all necessary operators.
- Finding the number-conserving ($2 \rightarrow 2$) set of possible many-body transitions between the basis states
- Calculating matrix elements for the $q\bar{q}$ excitation potential, if applicable, and the table of triples associated with it.
- Finding the additional ($1 \leftrightarrow 3$) many-body transitions which do not conserve the number of particles.
- Assembling the elements of the Hamiltonian.
- Applying the Lanczos algorithm to tri-diagonalize the Hamiltonian.
- Finding the eigenvalues and eigenvectors of the Lanczos matrix.
- Performing density matrix calculations to obtain the quantum numbers of the eigenstates.

Each one of these can be implemented as a separate program module which reads in data from files and produces further output for use at a later stage. It should be noted that a single pass through the above cycle will only give us one set of eigenvalues for a particular value of the two-centre separation, Z as discussed in section 3.1. In practice we want to accumulate a set of enough data points to give us a graph of the interaction, so the whole process is wrapped within a loop over a range of Z values specified in the initial data file and the calculation is performed for each value.

Not all of the steps need to be repeated when the input data has been altered (for example, if only the separation is changed it isn't necessary to re-build the table of basis states and the sets of pairs and triples), so the program always stores a log of the details of its previous calculation and consults this before proceeding, enabling it to skip unnecessary steps and reuse the data files from the previous run.

We will first discuss the starting data required for the program and then follow the process through in sequence, providing skeleton sections of code (in 'C') to illustrate each stage. Details such as memory management, allocation of arrays, file handling, function templates and global data declarations have been omitted.

C.1 Program Input

The initial data can be split into roughly three parts

- Data defining the model space
- Parameters needed to calculate the matrix elements
- Program control parameters and flags

C.1.1 Model Space Data

The model space is determined by the shells we are using, their occupancies, the values of M_s, M_t and parity, and the minimum (`MINQ`) and maximum (`MAXQ`) number of quarks in the calculation, all of which are inputs to the

program. By "shells" in this case, we mean the spatial wavefunctions ϕ_+ and ϕ_- defined in section 3.2. Distinct shells are considered for quarks and antiquarks. A full set, with occupancy values for a full calculation is shown in table C.1. So in this case we have a maximum seven quarks and minimum of

Shell	Spatial Parity	Intrinsic Parity	Min. Occ.	Max. Occ.
$\phi_+(q)$	0	0	0	7
$\phi_-(q)$	1	0	0	7
$\phi_+(\bar{q})$	0	1	0	1
$\phi_-(\bar{q})$	1	1	0	1

Table C.1: Shells and Occupancies

six. They are all allowed to move freely throughout the space so the minimum occupancy for each quark shell is zero and the maximum is 7. Similarly, the occupancies for the antiquark shells are a minimum of zero and maximum of one.

C.1.2 Model Parameter Data

These are basically just the model parameters discussed in section 5.2. The constant values of C, m, α_s and a_c are all read in and used for matrix element calculations. A boolean flag value is used to ascertain whether a linear, rather than the default quadratic confinement potential should be used. The range of Z values is also specified, in the form of minimum and maximum values and an increment value to determine the size of the steps between them.

C.1.3 Program Control Parameters

The most important of these are the ones which affect the diagonalization of the Hamiltonian and the calculation of eigenstates. These are the number of required iterations, the desired number of eigenstates and a flag to determine whether a starting vector obtained from a previous calculation should be used (see section 3.10.2). It should be remembered that the Lanczos algorithm gives most rapid convergence for the extremal eigenvalues, so more iterations may be needed if a larger number of eigenstates are requested. The program has

a safety mechanism built in to check for convergence of the eigenvalues, and more iterations will be performed if the states have not fully converged.

In addition, several flags can be used to override the program's decisions on which parts it needs to carry out, based on it's log of data from the previous full calculation performed. For example, the computer on which the program is running may crash while our calculation is busy diagonalizing the Hamiltonian. By this stage the Hamiltonian will have been assembled and stored and we would not wish to have to repeat the stages leading up to this, so we can set control flags to indicate that the stored data files should be used and the calculation should re-commence at the Lanczos algorithm routine.

C.2 Single Particle Orbits

This simple routine merely constructs an array of the orbits and their quantum numbers. Here, as elsewhere in the program, half-integer quantum numbers are doubled to ease handling. The spatial (`n`) and intrinsic (`p`) parities of each shell in the calculation are read in and combined with z-components of spin and isospin, giving four quantum numbers for each orbit. Strangeness and mass would also be added for calculations involving strange quarks.

The orbits can either be set up as 'C' structures or as separate quantum number arrays indexed by the orbit number. We will assume the former. So at the end of the routine we have a single array, `orbits[nOrbits]`, each element of which is a structure with four fields `p, n, ms, mt`.

C.3 Many-Particle Basis

Constructing the many particle basis states can be split into two main sub-processes:

- Finding the valid single-colour SDs. We usually refer to these as "red states".
- Combining these in threes to make up the actual basis.

as discussed in section 3.3.4.

C.3.1 Generating the Red States

The red states are found by means of the bit-shifting mechanism in section 3.3.4. This is performed for each valid number of red particles. *i.e.*,

```
int nRed;
int minRed = MINQ/3;
int maxRed = minRed + 2*(MAXQ-MINQ);

for(nRed = minRed; nRed <= maxRed; nRed += 2)
{
   generateRedStates(nRed);
}
```

The shifting is done by recursive calls to a particle moving function. We start with the lowest `nRed` orbits filled and attempt to move the first particle to the next orbit. If the orbit above is filled, the move function calls itself recursively for subsequent particles, until it either succeeds in moving one or the last particle is reached. In the latter case the process terminates. The number of quarks and antiquarks in the red state resulting from each valid move are noted and, if acceptable, the binary representation of the red state is stored in an array, `redStates[nRedStates]`, ready for use in the combination process. It is important to realise that this procedure generates an ordered list of red states. When interpreted as binary numbers `redStates`$[i]$ <`redStates`$[i+1]$, for all i in the range $1,\ldots,$`nRedStates`.

C.3.2 Combining the Red States

The red, green and blue codes are selected from the `redStates` array by means of three nested loops of indices. So we have:

```
int R, G, B;
for(R = 0; R < nRedStates; R++)
{    ...
     for(G = R; G< nRedStates; G++)
     {    ...
          for(B = G; B < nRedStates; B++)
          {  // check combined quantum numbers here.
             // If OK, write the three codes and
             // multiplicity to the basis file
```

The quantum numbers of each colour code are calculated from its orbits at the start of each loop and used to work out those of the overall state. If these match the spin, isospin and parity values from the input data, the state information is written to a data file in the form of the three colour codes and the multiplicity of the state. Note the starting values of the indices `R,G,B` of the loops impose the ordering of the colour codes R, G, B in a three-colour SD: $R \leq G \leq B$.

C.4 The Pair Sets

In this part of the program, we assemble the sets of pairs of orbits with the same quantum numbers, which may have non-zero matrix elements of the two-body Hamiltonian, as discussed in section 3.4.1. The two-body transitions must conserve M_s, M_t overall parity and spatial and intrinsic parity separately, so all pairs in a set must satisfy these conditions. The quantum numers of the individual particles in a pair may be changed by the action of the Hamiltonian but the sums must remain unchanged. The colour of the particles is also important, and the sets are also differentiated by whether the two particles in the pair have the same or different colour. When the particles have different colour, we must also differentiate between two-particle states of the form $\mid q\bar{q}\,\rangle$ and $\mid \bar{q}q\,\rangle$ because we cannot have e.g. $r\bar{g} \rightarrow \bar{r}g$. Separate sets will also be used for these different pair types. For the $q\bar{q}$ case when the colours are the same, the possibility of $r\bar{r} \rightarrow g\bar{g}$ and $r\bar{r} \rightarrow b\bar{b}$ must be considered as well as $r\bar{r} \rightarrow r\bar{r}$. This is done by adding an extra pair into the set, marked by an extra large packing value (see below). Later code is designed to notice this and include the extra transitions and different matrix elements as appropriate.

The sets are obtained by a series of nested loops giving values for c (logical colour) [1], M_t, M_s, spatial parity and intrinsic parity. There is also an extra loop which comes into play for the $r\bar{g}$ case discussed above.

Within these, a further two loops define the two orbit numbers of the pair. Each possible pair is tested to see whether it has the correct quantum numbers for the set. If it does, the pair is stored in an array (`pairs`) (the orbits are

[1] See section 3.4.1. Logical colour 0 means colours are the same, 1 means they are different

packed into one number to enable this) and the set number of the pair is stored in another array, using equations 3.4.4 to calculate the index. The code would look like this

```
#define PACK 256

void pairSets(void)
{
     int i,j,imax,jmin;
     int np, tnp=0;              // tnp = total number of pairs
     int c,mt,mj,p,ps,index,extra,set=0,extraSet,p_of_i;

     nPairs = 0;
     for(c = 0; c <= 1; c++)      // colours the same/different
        for(mt=-2; mt<=2; mt+=2)
        {
           for(mj=-2; mj<=2; mj+=2)
              for(ps=0; ps <=1; ps++)    // spatial parity
                 for(p=0; p <= MAXQ-MINQ; p++)  // intrinsic Parity
                 {
                    extraSet = p && c ? 1 : 0;
                    for(extra=0; extra <= extraSet; extra++)
                    {
                       p_of_i = -99; // intrinsic parity of orbit i
                       np = 0;
                       iMax = c ? nOrbits : nOrbits-1;
                       for(i = 1; i <= iMax; i++)
                       {
                          jMin = c ? 1 : i+1;
                          for(j = jMin; j <= nOrbits; j++)
                          {
                             if(orb[i].mj + orb[j].mj != mj
                                  || orb[i].p + orb[j].p != p
                                  || orb[i].f + orb[j].f != f
                                  || orb[j].n != (ps + orb[i].n)%2)
                                        continue;
                             if(c == 1 && p_of_i == -99)
                                  p_of_i = orb[i].p;
                             if(c == 1 && orb[i].p != p_of_i)
                                  continue;
                             if(c == 0)
                               index=((2*nOrbits-i)*(i-1))/2 + j-i;
                             else
```

```
                        index=(nOrbits*(nOrbits + 1))/2 +
                                        nOrbits*(i-1) + j;
                    pairSet[index] = set + 1;
                    np++;  tnp++;
                    pairs[tnp] = PACK*i + j;
    // add extra pair for different coloured mesons i.e. <11|v|00>
                    if(c == 0 && p == 1)
                    {
                       np ++; tnp ++;
                       pairs[tnp]=PACK*PACK + PACK*i + j;
                    }
                 }
              }
              if(np != 0)   // if the set isn't empty
              {
                 pairsInSet[ ++set ] = np;
                 startPairs[set] = nPairs + 1;
                 nPairs += np;
              }
           }
        }
     }
     if(c == 0)
        lastOset = set;
  }
  nSets = set;
}
```

where we have assumed that the arrays which store the properties of the sets are defined globally. These are

`pairs` – the array of packed pairs of orbit numbers.

`pairSet` – the set number for a particular pair.

`pairsInSet` – the number of pairs in a particular set.

`startPairs` – the position in the pair array where a particular set of pairs begins.

Other global variables which are used later are the number of pairs (`nPairs`), the number of sets (`nSets`) and the last set with $c = 0$ (`lastOset`).

C.5 Two-Body Matrix Elements

This routine works its way through all the pairs in each set and works out all matrix elements of the form $\langle kl \mid v_{12} \mid ij \rangle$ for all the two-body operators in the model, using the formulae from chapter 4. The routines for assembling the Hamiltonian and for density matrix calculations use the matrix elements from this section. In the sections discussing these, we will assume that the matrix elements for the potential are stored in an array called `reals[]` with an another array called `startReals[]` giving the position of the elements for each pair. The elements for additional operators are stored in other appropriately named arrays.

C.6 Many-Body Transitions

Here we use convert the two-body transitions (which we have stored in terms of our sets of orbit pairs) into the transitions between the actual basis states of the model, enabling us to construct the many-body Hamiltonian. This routine takes each basis state in turn, and destroys all possible pairs of particles in it. For each destroyed pair, it locates the set in which the pair occurs (by consulting the array `pairSet`) and creates all possible pairs in the same set. Each new SD thus created is searched for in the table of basis states and its number is stored in a file along with a pointer to the matrix element associated with the transition (*i.e.* an index into the `startReals` array).

The pair destruction and creation for each basis state is divided into two separate routines – one for pairs of the same colour, and one for those of different colour. So the basic form of the routine would look like this

```
#define Red 0
#define Green 1
#define Blue 2

void manyBodyHamilton(void)
{
     int state, zero = 0;

     for(state = 1; state <= basisSize; state++)
     {
```

```
    // write the state number to the file
        fwrite(&state, sizeof(int),1,componentfile);
        destroySameColour(state);
        destroyDifferentColour(state);
    // write a zero to mark the end of transitions for this state
        fwrite(&zero,sizeof(int),1,componentfile);
    }
}
```

In what follows, we also assume that the basis of SDs has been loaded into a two dimensional array `basis[3][basisSize]`, indexed by the state number and the colour (`Red = 0, Green = 1, Blue = 2`). The elements give the binary representation of the orbits of the specified colour in the given state. For each cycle of the loop, the state being acted on is further broken down – the details of the SDs particles are stored in an array called `orbits[3][]` indexed by the particle number and colour and the number of particles of each colour is stored in an array called `nParticles[3]`. At this stage, we use the unpacked orbits to form the parity representation of the state (stored in `parityRep[3]`), for calculating the phase of the operators (see section 3.7). The elements are formed using the following code, for each value of the variable `colour`

```
int i, code=0;

for(i = 1; i <= nParticles[colour]; i++)
    code^= (1 << orbits[colour][i]) - 1;
parityRep[colour] = code;
```

With these extra details, we are ready to move onto the routines which do the annihilation and creation of pairs, converting the state into others in the basis.

C.6.1 Same-Colour Pair Annihilation

The code for the same-colour routine looks something like this

```
void destroySameColour(int state)
{
    int i,col,co,creCol; // colour indices
    int dphase,cphase; // phases for destruction and creation ops
    int part1,part2;   // destroyed pair particle numbers
```

```
int orb1,orb2;      // the destroyed orbit pair
int crorb1,crorb2; // the created orbit pair
int pairIndex, set;
int changingColour; // boolean value
int pr,minPr,maxPr;
int newbin;      // binary rep of created orbits

for(col = Red; col <= Blue; col++)
{
    for(part1 = 1; part1<=nParticles[col]-1; part1++)
    {
        orb1 = orbits[col][part1];
        dphase = 1;
        for(part2 = part1+1; part2<=nParticles[col]; part2++)
        {
            orb2 = orbits[col][part2];
            for(co = Red; co <= Blue; co++)
            {
                residue[co] = basis[co][state];
                parResidue[co] = parityRep[co];
            }
            residue[col] ^= (1<<orb1) + (1<<orb2);
            parResidue[col] ^= (1<<orb1) - 1;
            parResidue[col] ^= (1<<orb2) - 1;
            pairIndex = ((2*norbits - orb1)*(orb1-1))/2
                                        - orb1 + orb2;
            set = pairSet[pairIndex];
            minPr = startPairs[set];  // First Pair in the Set
            maxPr = pairsInSet[set] + p1 - 1;  // Last Pair
            realPtr = startReals[pairIndex]-1;
            for(pr = minPr; pr <= maxPr ; pr++)// Create pair pr
            {
                realPtr ++;
                crorb1 = pairs[pr]/PACK;
                crorb2 = pairs[pr]%PACK;
                changingColour = crorb1 > PACK;
                if(changingColour)
                    crorb1 = crorb1%PACK;
                newbin = (1 << crorb1) + (1 << crorb2);
                creCol = col;
                for(i = 1; i <= (changingColour ? 2 : 1); i++)
                {
                    if(changingColour)
```

```
                        creCol = (creCol + 1)%3;
 /* If these orbits */  if((residue[creCol] & newbin) == 0)
 /* aren't occupied */  {
                        if((1 << crorb1) & parResidue[creCol])
                            cphase = -1;
                        else
                            cphase = 1;
                        if((1 << crorb2) & parResidue[creCol])
                            cphase = -cphase;
                       // Create the new state and locate
                       // its position in the basis. Write
                       // the details to the output file.
                       // ....
                        locate(newState, dphase*cphase);
                      }
                    }
                }
                dphase = -dphase;
            }
        }
    }
}
```

The code selects each single-colour SD for the states in question and steps through all the pairs of particles in it. The orbits of the destroyed pair are removed from both the occupancy and parity representations of the state and the results stored in the arrays `residue` and `parResidue` respectively. The orbit numbers of the destroyed particles are used with equations 3.4.4 to work out an index for the pair, giving us a handle on the set which the pair belongs to and its associated information, as stored by the pair-finding routine discussed above in section C.4. Each pair in the set is then created in turn (the `for` loop over the variable `pr`). The orbits are unpacked from the pair array, checking for the colour-changing case ($r\bar{r} \rightarrow g\bar{g}, r\bar{r} \rightarrow b\bar{b}$) and a binary representation of the created orbits is formed (`newbin`). We then proceed to check the new state is valid (*i.e.* the created orbits aren't already occupied) and work out the phase for the creation of the two particles as detailed in section 3.7.

The variable `creCol` gives the colour of the created pair which will normally be in the same single-colour SD as the destroyed pair. In the changing-colour case, though, we have to create it in each of the other two SDs, hence the

inclusion of the loop over i, which is performed once or twice as necessary.

The new state is located in the basis array by means of a standard binary search over its three single-colour SD values. It is possible that the created state may be a ghost, in which case the red, green and blue SD values must be rearranged into ascending order prior to the search. The number of the new state and the index (realPtr)and phase factor of the associated matrix element for the transition are stored to disc.

C.6.2 Different-Colour Pair Annihilation

The code for this part has essentially the same form as the same-colour case. There are a few obvious changes to deal with the fact that the created and destroyed particles are now in different single-colour SDs. These SDs have been labelled red, green and blue, for convenience. The actual colours vary, but that isn't important.

```
void destroyDifferentColour(int state)
{
    int red,green,blue; // the SDs
    int pairIndex,set;
    int part1,part2;
    int col1,col2;
    int orb1,orb2;
    int crorb1,crorb2;
    int pr,minPr,maxPr;
    int newbin1, newbin2;
    int redPhase,greenPhase,dphase1,dphase2,cphase;

    for(col1 = Red; col1 <= Green; col1++)
    {
        for(col2 = col1+1; col2 <= Blue; col2 ++)
        {
            blue = basis[3-col1-col2][state];
            dphase1 = 1;
            for(part1 = 1; part1 <= nParticles[col1]; part1++)
            {
                orb1 = orbits[col1][particle1];
                red = basis[col1][state] - (1 << orb1);
                redphase = parityRep[col1] ^ ((1 << orb1) - 1);
                dphase2 = 1;
                for(part2 = 1; part2 <= nParticles[col2]; part2 ++)
```

```
                {
                    orb2 = orbits[col2][part2];
                    green = basis[col2][state] - (1 << orb2);
                    greenPhase = parityRep[col2] ^ ((1<<orb2)-1);
                    pairIndex = nOrbits*(nOrbits+1)/2
                              + nOrbits * (orb1-1) + orb2;
                    set = pairSet[pairIndex];
 /* First Pair */   minPr = startPairs[set];
 /* Last Pair  */   maxPr = pairsInSet[set] + minPr - 1;
                    realPtr = startReals[pairIndex] - 1;
                    for(pr = minPr; pr <= maxPr; pr ++)
                    {
                        realPtr ++;
                        crorb1 = pair[pr]/pack;
                        crorb2 = pair[pr] % pack;
                        newbinRed = 1 << crorb1;
                        if((red & newbinRed) != 0)
                            continue;
                        newbinGreen = 1 << crorb2;
                        if((green & newbinGreen) != 0)
                            continue;
                        newState[Red] = red + newbinRed;
                        newState[Green] = green + newbinGreen;
                        newState[Blue] = blue;

                        if(redPhase & newbinRed)
                            cphase = -1;
                        else
                            cphase = 1;
                        if(greenPhase & newbinGreen)
                            cphase = -cphase;
                        locate(newState,dphase1*dphase2*cphase);
                    }
                    dphase2 = -dphase2;
                }
                dphase1 = -dphase1;
            }
        }
    }
}
```

C.7 Quark-Antiquark Excitations

The part of the calculation implementing the Yu-Zhang $(1 \rightarrow 3)$potential into the model bears a strong resemblance to the parts of the program which have already been described for the $2 \rightarrow 2$ case. The arrays `triples`, `startTriples`, `triplesInSet` function in much the same way as the corresponding arrays for the sets of pairs. The main difference in the approach is that the array of matrix elements for the potential is calculated in the same routine which finds the sets of triples. In many ways, the process is simpler than that for the pairs – we are only destroying one particle, so the number of sets is just equal to the number of orbits, and we fill each set with the triples which match the quantum numbers of the destroyed orbit.

The code is shown below and follows the same sort of structure as the pair-finding routine. There is a loop for the destroyed orbit, `i`, and then three nested loops for the created orbits `j`,`k` and `l`. In all cases, at least one of the created particles will have the same colour as `i` so we assume that `j` always has the same colour. If `j` happens to be the odd particle [2], we must take care not to allow transitions in which `k` and `l` do not have the same colour as `i`, because $r- > \bar{r}gg$ etc. are not permitted. If the particles are all the same colour, then the `k` loop no longer starts from 1 but begins at $j + 1$. This is to prevent identical triples being generated more than once. The three created orbits are packed into one number and then stored in the triple array.

```
#define triple(a) ( orb[j].a + orb[k].a + orb[l].a )

void YuZhang(void)
{
    int i,j,k,l; // orbit numbers
    int cj,ck,cl; // logical colour values
    int jOdd;  // boolean - true if j is the 'odd' particle
    int realPtr=0;
    int tnt=0,nt; // triple counters

// Destroy loop
    cj = 0;   // j always has the same colour as i
    for(i = 1; i <= nOrbits; i++)
```

[2]For single meson creations, the odd particle is just the antiquark. We keep the case general, though, to allow $\bar{q} \rightarrow q\bar{q}\bar{q}$.

```
    {
        nt = 0;
        startTriples[i] = tnt + 1;
// Create loops j,k,l
        for(ck = 0; ck <= 1; ck++)
        {
            cl = ck;      // the other two must also be same colour
            for(j = 1; j <= nOrbits; j++)
            {
                jOdd = (orb[i].p != orb[j].p);
                if(jOdd && (ck == 1))
                    continue;
                for(k = (ck == 0 ? j+1 : 1); k <= nOrbits; k++)
                {
                    for(l = k+1; l <= nOrbits; l++)
                    {
                        if(triple(p) != orb[i].p + 1
                           || triple(n)%2 != (orbits[i].n + 1)%2
                           || triple(mt) != orbits[i].mt
                           || triple(mj) != orbits[i].mj)
                                continue;
                        triples[++tnt] = 1000000*(10*j+cj)+
                                         1000*(10*k+ck)+(10*l+cl);
                        nt++;
                        YZreals[++realPtr] =
                              conjugateParticle(i,j,k,l,cj,ck,cl);
                    }
                }
            }
        }
        triplesInSet[i] = nt;
    }
}
```

The function `conjugateParticle` performs the conjugation of the odd particle to allow the matrix elements to be calculated in two-body form (see section 4.3.1). It assumes particle `l` is the odd particle and, if necessary, `j` or `k` are swapped with it to ensure that this is the case. It also calls the finds the matrix element associated with the transition, which is then stored in the array `YZReals`.

```
float
```

```
conjugateParticle(int i,int j,int k,int l,int cj,int ck,int cl)
{
    int pl,mtl,nl,mjl;
    float Vdir,Vexc,Cdir,Cexc,lphase;

    if(orbits[j].p != orbits[i].p) // j odd
    {
        swap(l,j);
        swap(cl,cj);
    }
    else if(orbits[k].p != orbits[i].p)  // k odd
    {
        swap(l,k);
        swap(cl,ck);
    }
    pl = 1 - orbits[l].p;   // Conjugate quantum numbers
    mtl = -orbits[l].mt;
    nl = orbits[l].n;
    mjl = -orbits[l].mj;
    if(pl == 0)
        lphase = pow(-1,(1 - (mjl + mtl)/2));
    else
        lphase = pow(-1,(1 + (mjl + mtl)/2));
    Vdir = YZspace_spin(j,k,i,mtl,nl,mjl);
    Vexc = YZspace_spin(k,j,i,mtl,nl,mjl);
    Cdir = colval[0][cj][ck][0][cl]/4;  // <j,k| L1.L2 |i,l>
    Cexc = colval[0][ck][cj][0][cl]/4;
    return (Vdir * Cdir - Vexc * Cexc) * lphase;
}
```

C.7.1 Many-Body Hamiltonian

The assembly of the $1 \leftrightarrow 3$ part of the many-body Hamiltonian is virtually the same as the $2 \rightarrow 2$ case, discussed above in section C.6. For each state in the basis, a single particle is annihilated and the triples from the corresponding set are created in turn, locating the resulting new states in the basis and storing the details as before.

C.8 Building the Hamiltonian

This routine is concerned with assembling the elements of the Hamiltonian from the many-body data accumulated in the previous sections. This is done a row at a time, starting from the diagonal element. There are essentially two sets of input data – one for the number-conserving part of the interaction (section C.6) and one from the transitions relating to the Yu-Zhang interaction (section C.7.1). Each of these data files consists of a set of lists of "components" for each state (*i.e.* row) in the basis. Each component consists of the number of another state in the basis (*i.e.* the column) and the matrix element and phase factor for the transition between the two basis states. To assemble a row, we just have to add up the effects of all the components. There are a few other bits and pieces to be dealt with, such as the single particle energies and the multiplication factors mentioned in section 3.12 which arise from our treatment of the ghost states. Even so, the code is reasonably simple.

```
static float sqrtFactor[3] = { 1.0, SQRT3, SQRT6 };
int YZRow = 0;
extern FILE *Hamiltonfile,  // the output file
            *componentfile; // many body data file
            *YZFile;        // the Yu-Zhang components

void makeHamiltonian(H_elt *buffer)
{
    int j, row;
    int state;
    int nParticles; // number of particles in state
    int Mass;       // sum of particle masses in state
    int stateOrbits[10]; // array to store orbits in state
    float kfactor, spe;  // spe = single particle energy

    kfactor = 10*hbar_c_MeV_fm * hbar_c_MeV_fm /(C * C);
    for(row = 1; row <= basisSize; row++)
    {
        fread(&state,sizeof(int),1,componentfile);
        // assign nParticles and Mass here ...
        // load stateOrbits[] with the orbit numbers of the state
        spe = 0;
        for(j = 1; j <= nParticles; j++)
        {
            int o = stateOrbits[j];  // Single Particle KE Operator
```

```
            spe += orbits[o].p2*(1.0/orbits[o].m - 1.0/Mass);
        }
        spe *= 0.5*kfactor;
        spe += Mass;
        element[row] = spe;          // diagonal element
        makeRow(row, buffer, kfactor/Mass);
    }
}
```

We have assumed that the single particle kinetic operator matrix elements are stored in the p2 field of the orbit structure. An additional structure type called `H_elt` has also been defined to hold store the elements of the Hamiltonian. It has two fields – a floating point number (`elt`) for the numerical value and an integer (`index`) to store the column number. The following code is used to build each row.

```
void makeRow(int row, H_elt *buffer,float kfoverM)
{
    int eltsInRow,j,state,column,nComponents=0,phase,rpt;
    H_elt *startBuffer = buffer;
    fread(&state, sizeof(int), 1, componentfile);
    while(state != 0)
    {
        if(state < 0)
        {
            phase = -1;
            state = -state;
        }        // elt[state] is <state|H|row>
        else
            phase = 1;
        rpt = state & 0x7fff; // unpack state and rpt
        state >>= 15;
        elt[state] += phase*(reals[rpt] +
            p1p2[rpt]*kfoverM);      //  Two body KE operator
        columnOf[ ++nComponents ] = state;
        fread(&state, sizeof(int), 1, componentfile);
    }
    if(YZRow == row)
    {
        fread(&state, sizeof(int), 1, YZFile);
        fread(&rpt, sizeof(int), 1, YZFile);
        while(state != 0)
```

```
        {
            if(rpt < 0)
            {
                phase = -1;
                rpt = -rpt;
            }
            else
                phase = 1;
            elt[state] += yzfactor * YZreals[rpt] * phase;
            columnOf[ ++nComponents ] = state;
            fread(&state,sizeof(int),1,YZFile);
            fread(&rpt,sizeof(int),1,YZFile);
        }
        fread(&YZRow,sizeof(int),1,YZFile);
    }
    buffer -> index = 0;
    buffer++ -> elt = 0;
    buffer -> index = row;
    buffer++ -> elt = elt[row];    // Diagonal Element
    eltsInRow = 2;
    elt[row] = 0.0;
    for(j = 1; j <= nComponents; j++)
    {
        column = columnOf[j];
        if(elt[column] != 0.0) // Store non-zero elts and their
        {                      // position (column)
            eltsInRow++;
            buffer -> index = column;
            buffer++ -> elt = elt[column]*sqrtFactor[mplty[row]/3]
                                    / sqrtFactor[mplty[column]/3];
            elt[column] = 0.0;
        }
    }
    fwrite( startBuffer, sizeof(H_elt), eltsInRow, Hamiltonfile);
}
```

The components for the row are read in and the matrix element for each is added to the appropriate element of the array `elt[]`. This is then repeated for the Yu-Zhang data. It would then be possible to go through the row a column at a time and write the accumulated elements to the Hamiltonian data file. However, because of the sparsity of the matrix, it is found in practice that the number of components is smaller than the size of the basis, so it is faster

to find the columns which have non-zero matrix elements by looping through these components instead. To do this, a counter is kept of the number of components and the column of each is kept in a separate array `columnOf[]`. In this way, all the non-zero elements of the row are found, their column position and value are written to a buffer where they are stored prior to being written to the Hamiltonian file. The value is first multiplied by the weighting factor $(\sqrt{m_i}/\sqrt{m_j})$ from section 3.12. The array `mplty[]` holds the multiplicities of the states, as stored in the basis file.

C.9 Tri-Diagonalizing the Hamiltonian

We are now in a position to apply the Lanczos algorithm and find the eigenvalues and eigenvectors of the Hamiltonian. The Lanczos algorithm is widely used and details of its application are readily available. Nevertheless, we will discuss it briefly because it is such a central feature of the Glasgow shell-model code.

This part of the program is by far the most time consuming. The multiplication of the Lanczos vectors by the Hamiltonian matrix, the normalization, orthogonalization and re-orthogonalization of vectors are all very labour intensive processes. The latter obviously becomes more important as the diagonalization proceeds and the number of Lanczos vectors increases. Unfortunately the scope for heavy optimization is somewhat limited and would rely mainly on standard methods for matrix and vector operations. These might include loop unrolling, where more work is done in each cycle of a loop to avoid the pipeline overheads caused by comparison operations in the code. Matrix multiplications can be optimized to make the most efficient use of on-chip and secondary-cache memory to avoid the significantly larger times needed to fetch data from conventional memory. Such optimizations can drastically improve performance of procedures such as matrix-vector multiplications and dot products. However, to a large extent they are dependent on the architecture being used and the exact details of the implementation are often best established by trial and error (using a time-profiling compiler option to establish the amount of time taken in functions) rather than by theoretical means. Moreover, they are by no means specific to our application so we will not pursue them here.

We also have to read and write vectors and parts of the Hamiltonian to disc throughout the process, so this must also be done as efficiently as possible. This will obviously depend on what memory resources are available. We present here a generic implementation of the Lanczos algorithm to which further machine-specific optimizations may be added as required. The main function looks like this

```
#define BUFFERSIZE 200000

void LanczosIterations(float *alpha, float *beta, H_elt *H)
{
    int it,k;
    float a,*v0,*v1,*v2,*vtemp;

    v0 = zeroVector(1,basisSize);   // allocate vectors
    v1 = zeroVector(1,basisSize);
    v2 = vector(1,basisSize);
    beta[0] = 0;
    setFirstVector(v1);
    normalize(v1,basisSize);
    writeVector(v1,lancfile,basisSize);
    for(it = 1; it <= nIterations; it++)
    {
        H_mult(v2,H,v1);    // v2 = H.v1
        alpha[it] = orthogonalize(v1,v2);
        addVectors(v2, -beta[it-1], v0, 1, basisSize);
        beta[it] = normalize(v2,basisSize);
        if(it > 1)
        {
            a = k = 0;
            while(a < 0.1 && k++ < 3)
            {
                if (it == basisSize)
                    break;
                a = reorthogonalize(v2,v1,it);
                if(k > 1)
                    printf("Re-reorthogonalizing, %d\n",it);
            }
            if(a < 0.1 && it < basisSize)
            {
                printf(" Reorthogonalization Failed\n\a");
                break;
```

```
            }
        }
        writeVector(v2,lancfile,basisSize);
        vtemp = v0;     // arrange vectors for next iteration
        v0 = v1;
        v1 = v2;
        v2 = vtemp;
    }
    beta[nIterations] = 0;
    fclose(Hamiltonfile);
}
```

The arguments passed are vectors to store the α and β values as introduced in section 3.10 and a pointer to the buffer (as used in the previous section) in which the elements of the Hamiltonian are to be stored. First, memory for the Lanczos vectors is allocated (we need space for three vectors for each iteration) and a function is called to set the elements of the starting vector, as described in section 3.10.2. This is then written to the file of Lanczos vectors (`lancfile`).

The main loop of the procedure carries out the iterations, applying the equations from section 3.10 and generating the new Lanczos vectors. The middle block of code deals with the re-orthogonalization process, performed by the `reorthogonalize` routine which orthogonalizes $\boldsymbol{v}_2$ to each of the previous Lanczos vectors in turn. At this stage the value of the norm of the new $\boldsymbol{v}_2$ is checked to ensure that it is not too close to zero. $\boldsymbol{v}_2$ is initially normalized before re-orthogonalization, but if the accumulation of rounding errors leads to a vector being generated which is almost spanned by the previous vectors, this will be shown by a low norm value for the new, re-orthogonalized vector. If it is too low, the size of the vector may have been reduced to the extent that we may not be able to rely on its orthogonality when rounding errors are taken into account. An attempt to redress the situation is made by repeatedly re-orthogonalizing up to three times until a norm larger than 0.1 is found. Otherwise the algorithm terminates, but this never actually happened in practice.

Once a satisfactory vector $\boldsymbol{v}_2$ has been obtained, it is stored and becomes the next $\boldsymbol{v}_1$, $\boldsymbol{v}_1$ becoming the next $\boldsymbol{v}_0$ ready for the following iteration.

The routine which performs the matrix multiplication $v_2 = Hv_1$ is shown

below, along with the orthogonalization and re-orthogonalization functions. They are largely self-explanatory. The function `normalize` is also assumed to exist. This just normalizes the vector argument and returns the value of the norm (given by $\sqrt{v \cdot v}$).

```
static void H_mult(float *v2, H_elt *buffer, float *v1)
{
    int i, buffelts, row = 0, column;
    float element;
    H_elt *start_buffer = buffer;

    setVector(v2, ZERO, 1, basisSize);

    rewind(Hamiltonfile);
    while( !feof(Hamiltonfile) )
    {
        buffer = start_buffer;
        buffelts = fread(buffer,sizeof(H_elt),
                         BUFFERSIZE,Hamiltonfile);
        for(i = 0; i < buffelts; i++)
        {
            column = buffer -> index;
            element = buffer++ -> elt;
            if(column == 0)     // new row
            {
                row++;
                i++;
                v2[row] += buffer++ -> elt * v1[row];
            }
            v2[row] += element * v1[column];
            v2[column] += element * v1[row];
        }
    }
}

double orthogonalize(float *v1,float *v2)
{  // orthogonalize v2 to v1
    int i;
    double scalar=0;
    for(i=1; i <= basisSize; i++)  // Dot Product
        scalar += v1[i] * v2[i];
    for(i=1; i <= basisSize; i++)
        v2[i] -= scalar * v1[i];
```

```
    return(scalar);
}

double reorthogonalize(float *v2,float *v1,int nVecs)
{
    int vec;
    rewind(lancfile);
    for(vec = 1; vec <= nVecs; vec++)
    {
        readVector(v1,lancfile,basisSize);
        orthogonalize(v1,v2);
    }
    return(normalize(v2,basisSize));
}
```

The orthogonalization and normalization routines would probably benefit substantially from the loop-unrolling procedure mentioned above. In this case, we would have something along the lines of

```
    for(i=1; i <= basisSize; i += n)
    {
        scalar += v1[i] * v2[i]
                + v1[i+1] * v2[i+1]
                + v1[i+2] * v2[i+2] + ... + v1[i+n]*v2[i+n];
    }
```

instead of the simple loop shown above in the orthogonalization function. Obviously additional loops would be required if `basisSize` was not an exact multiple of the constant `n`.

C.10 Diagonalizing the Lanczos Matrix

When the Lanczos algorithm has been completed, the tri-diagonal Lanczos matrix can easily be diagonalized using the methods described in section 3.11 and the references contained therein.

C.11 Density Matrix Calculations

The final stage is to analyse the composition of the eigenstates in terms of the number of particles and the quantum numbers for the overall state and for.

each cluster. This carries out the procedures from section 3.14 and subsequent sections.

```
void density(void)
{
    int i, j, nParticles, eigenState;
    int rpt;
    float evalue, exvalA;
    float element,*evector, *density, *mult;

    // allocate vectors here
    for(eigenState = 1; eigenState <= KEEP; eigenState++)
    {
        readVector(evector, eigenfile, basisSize);
        evalue = evector[0];
        for(j = 1; j <= nReals; j++)
            density[j] = 0.0;
        exvalA = 0.0;
     // Now form the density matrix elements
        for(i = 1; i <= basisSize; i++)
        {
            fread(&j,sizeof(int),1,componentfile);
            exvalA += nParticles * evector[i]*evector[i];
            fread(&j,sizeof(int),1,componentfile);
            while(j != 0)
            {
                if(j < 0)
                {
                    element = -1;
                    j = -j;
                }
                else
                   element = 1;
                rpt = j & 0x7fff;
                j >>= 15;
                element *= (evector[i] * evector[j]) *
                   (sqrtFactor[mult[i]]/sqrtFactor[mult[j]]);
                if(i != j) // Off diagonal elements
                    element *= 2;
                density[rpt] += element;
                fread(&j,sizeof(int),1,componentfile);
            }
        }
        printf("Eigenvalue %.4f:\n", evalue);
```

```
            quantumNumbers(density, exvalA);
        }
}
```

The above piece of code is basically a loop over the eigenstates which we want to examine (KEEP is the number of required states, specified in the starting data). The eigenvector (evector) is read in and the density matrix elements are calculated and stores in the array density. This process follows a similar line to the construction of the rows of the Hamiltonian, discussed above. The appropriate factors are added in to deal with the ghost states (sqrtFactor is assumed to be the same as in section C.8).

The function quantumNumbers then makes use of the density matrix for each state to calculate the expectation values of spin, isospin and colour operators for the overall state and also for the "half" state (*i.e.* the positive cluster). The latter part makes use of the projection operators from section 3.16. The variable N gives the value of the number-projection operator defined in section 3.16.1.

```
void quantumNumbers(float *density, float exvalA)
{
     int i;
     float Jwhole,Twhole,Cwhole,N,factor,Chalf,Thalf,Jhalf;

     Jwhole = Twhole = exvalA * (3.0/4); // One body terms
     Cwhole = exvalA * (16.0/3);
     N = 0.5 * exvalA;
     Jhalf = Thalf = 3*N;
     Chalf = 2 * Cwhole;  // 0.5 * 4
     for(i = 1; i <= nReals; i++)
     {
          factor = dens[i];
          Jwhole += Jelt[i] * factor;
          Twhole += Telt[i] * factor;
          Cwhole += Celt[i] * factor;
          N += Nprojection[i] * factor;
          Jhalf += Jprojection[i] * factor;
          Thalf += Tprojection[i] * factor;
          Chalf  += Cprojection[i] * factor;
     }
     Jwhole = 0.5 * (sqrt(1 + 4*Jwhole) - 1);
     Twhole = 0.5 * (sqrt(1 + 4*Twhole) - 1);
```

```
    Chalf /= 4;
    Thalf = 0.5 * (sqrt(1 + Thalf) - 1);
    Jhalf = 0.5 * (sqrt(1 + Jhalf) - 1);
    printf("J=%.2f T=%.2f C=%.2f\n",Jwhole,Twhole,Cwhole);
    printf("Half of State  N=%.2f J=%.2f T=%.2f C=%.2f\n",
                                N,Jhalf,Thalf,Chalf);
}
```

Bibliography

[1] B.A. Brown. New Vistas in the Nuclear Shell Model. *Nuclear Physics*, A522:221c–240c, 1991.

[2] B. Brown and B.H. Wildenthal. volume 38. Annual Reviews Inc., 1988.

[3] H. Yukawa. *Proc. Physc. Math. Soc. Japan*, 17:48, 1935.

[4] H. Yukawa. *Proc. Physc. Math. Soc. Japan*, 19:712, 1937.

[5] Robert Eisberg and Robert Resnick. *Quantum Physics of Atoms, Molecules, Solids, Nuclei, and Particles*, chapter 18, page 667ff. Wiley, second edition, 1985.

[6] D.H. Perkins. *Introduction to High Energy Physics*, chapter 8, page 283. Addison-Wesley, third edition, 1987.

[7] N. Isgur and G. Karl. *Phys. Rev.*, D20:1191, 1979.

[8] De Rújula, Georgi, and Glashow. Hadron masses in a gauge theory. *Phys. Rev.*, D12:147, 1975.

[9] D.A. Liberman. Short-Range Part of the Nuclear Force. *Phys. Rev.*, D16:1542, 1977.

[10] Frank E. Close. The Pauli Principle and QCD for Quarks and Nucleons in Hardons and Nuclei. In D. Fries and B. Zeitnitz, editors, *Quarks and Nuclear Forces*, chapter 2, pages 56–80. Springer-Verlag, 1982.

[11] Leila Ayat. *Spontaneous Creation of Quark-Antiquark Pairs in Few-Quark Systems.* PhD thesis, University of Glasgow, 1990.

[12] D.R. Hartree. *Proc. Camb. Phil. Soc*, 24:193, 1928.

[13] M. Weissbluth. *Atoms and Molecules*, chapter 19, page 385. Academic Press, 1978.

[14] M.G. Mayer. *Phys. Rev.*, (75):1968, 1949.

[15] M.G Mayer and J.H.D. Jensen. *Elementary Theory of Nuclear Shell Structure.* Wiley, 1955.

[16] P.J. Brussaard and P.W.M. Glaudemans. *Shell-Model Applications in Nuclear Spectroscopy*, chapter 16, page 327ff. North-Holland, 1977.

[17] Amos de Shalit and Igal Talmi. *Nuclear Shell Theory*, volume 14 of *Pure and Applied Physics*, chapter 9, page 52. Academic Press, 1963.

[18] I. Talmi. *Helv. Phys. Acta*, (25):185, 1952.

[19] R. Horgan. *Nuclear Physics*, B71:514, 1974.

[20] R. Horgan. *Nuclear Physics*, B129:45, 1977.

[21] M. Harvey and A.S. Jensen. *Nuclear Physics*, A179:33, 1972.

[22] A. Arima et al. In Michael Baranger and Erich Vogt, editors, *Advances in Nuclear Physics*, volume 5. Plenum Press, 1972.

[23] M. Born and J.R. Oppenheimer. *Ann. der Phys.*, 84:457, 1927.

[24] J.A. Wheeler. *Phys. Rev.*, 52:1083,1107, 1937.

[25] Makato Oka and Koichi Yazaki. Baryon-baryon interaction from quark model viewpoint. In W. Weise, editor, *Quarks and Nuclei*, pages 489–567. World Scientific, 1984.

[26] M. Harvey. Effective nuclear forces in the quark model with delta and hidden-colour channel coupling. *Nuclear Physics*, A352, 1981.

[27] S. Ohta, M. Oka, A. Arima, and K. Yazaki. A shell model study of six quark systems. *Physics Letters*, 119B:36–38, 1982.

[28] Marion H. Storm. *Quark Model Studies of the Nucleon-Nucleon Interaction.* PhD thesis, University of Glasgow, 1982.

[29] Marion H. Storm and A. Watt. Quark Model Contributions to Effective Nuclear Forces are Repulsive. *Nuclear Physics*, A408:397–416, 1983.

[30] G. Breit et al. *Phs. Rev.*, 126:881, 1962.

[31] F. Myhrer and J. Wrolsden. Nucleon-Nucleon Force and the Quark Degrees of Freedom. *Reviews of Modern Physics*, 60, 1988.

[32] R. Machleidt, K. Holinde, and Ch. Elster. *Phys. Rep.*, 149:1, 1987.

[33] A. Faessler. *Prog. Part. Nucl. Phys.*, 13:253, 1985.

[34] M. Oka and C.J. Horowitz. *Phys. Rev.*, D31:2773, 1985.

[35] G.Q. Liu, M. Swift, A.W. Thomas, and K.Holinde. The Role of Nuclear Structure in the NN interaction:Effects of Pion Exchange Between Quarks. *Nuclear Physics*, A556(3):331, 1993.

[36] Y. Fujiwara and K. T. Hecht. The NN Interaction in a Quark Model with Quark-Antiquark Excitations. *Nuclear Physics*, A444:541–578, 1985.

[37] S.D. Malik. *Shell Model Studies of the Quark-Antiquark Excitations in the Nucleon.* PhD thesis, University of Glasgow, 1989.

[38] K. L. Ayat and A. Watt. Quark-Antiquark Pair Creation Effects in the Study of $S = 0$ and $S = 1$ Baryons. *Nuclear Physics*, A535:592–604, 1991.

[39] R. R. Whitehead, A. Watt, B. J. Cole, and I. Morrison. The Glasgow Shell Model Code. In Michael Baranger and Erich Vogt, editors, *Advances in Nuclear Physics*, volume 9, chapter 2. Plenum Press, 1977.

[40] Amos de Shalit and Igal Talmi. *Nuclear Shell Theory*, volume 14 of *Pure and Applied Physics.* Academic Press, 1963.

[41] L.D. Landau and E.M. Lifshitz. *Quantum Mechanics: Non-Relativistic Theory*, volume 3 of *Course of Theoretical Physics*. Oxford, third edition, 1977.

[42] Mohammad Riaz. *Transputer Implementation for the Shell Model and SD Shell Calculations*. PhD thesis, University of Glasgow, 1990.

[43] Brian Ewins. *The Parallel Glasgow Shell Model Code and Applications*. PhD thesis, University of Glasgow, 1995.

[44] Brian W. Kernighan and Dennis M. Ritchie. *The 'C' Programming Language*. Prentice Hall, second edition, 1988.

[45] A.W. Roberts. *Elementary Linear Algebra*. Benjamin/Cummings, second edition, 1982.

[46] C. Lanczos. An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators. *Journal of Research of the National Bureau of Standards*, 45:255–282, 1950.

[47] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*, chapter 9, pages 322–351. Johns Hopkins University Press, 1989.

[48] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.

[49] A.R. Gourlay and G.A. Watson. *Computational Methods for Matrix Eigenproblems*, chapter 9, pages 79–81. Wiley, 1973.

[50] W.H. Press, S.A Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in 'C': The Art of Scientific Computing*. Cambridge University Press, second edition, 1992.

[51] James Ortega. The Given-Householder method for symmetric matrices. In A. Ralston and H.S. Wilf, editors, *Mathematical Methods for Digital Computers*, volume 2, chapter 4, page 96ff. Wiley, 1967.

[52] W.H. Press, S.A Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in 'C': The Art of Scientific Computing*, chapter 2, pages 44–55. Cambridge University Press, second edition, 1992.

[53] G. Breit. *Phys. Rev.*, 34:353, 1929.

[54] Y. Fujiwara and K. T. Hecht. $(q\bar{q})$ and $(q\bar{q})^2$ Excitations in a Quark Model of the NN Interaction. *Physics Letters B*, 171(1):17–21, 1986.

[55] Y. Fujiwara and K. T. Hecht. *Nuclear Physics*, A451:65, 1986.

[56] F. Myhrer and J. Wrolsden. *Physics Letters B*, 139:81, 1984.

[57] Y. Suzuki. Short-Range Part of the NN interaction. *Nuclear Physics*, A430:539–556, 1984.

[58] Zong-ye Zhang and You-wen Yu. Nucleon-meson vertex functions and isobar-meson vertex functions from the quark potential model. *Nuclear Physics*, A426:557–574, 1984.

[59] D. B. Lichtenberg. *Unitary Symmetry and Elementary Particles.* Academic Press, second edition, 1978.

[60] Frank E. Close. *An Introduction to Quarks and Partons.* Academic Press, 1979.

[61] Nathan Isgur. Nuclear Physics from the Quark Model. In Mikkel B. Johnson and Alan Picklesimer, editors, *Relativistic Dynamics and Quark Nuclear Physics.* Wiley-Interscience, 1986.

[62] M. Oka and K. Yazaki. In W. Weise, editor, *Quarks and Nuclei*, page 490. World Scientific, 1984.

[63] M. Harvey. *Nuclear Physics*, A352:301–325, 1981.

[64] Makato Oka and Koichi Yazaki. Short-range part of the nuclear force and the deuteron in a quark model. *Nuclear Physics*, A402:477–490, 1983.

[65] Makato Oka and Koichi Yazaki. Nuclear force in a quark model. *Physics Letters*, 90B:41–44, 1980.

[66] Makato Oka. Quark Model of the NN Interaction. In Donald F. Geesaman, editor, *Intersections Between Particle and Nuclear Physics*, May 1986.

[67] Koichi Yazaki. Nuclear force in the quark cluster model. *Progress of Theoretical Physics Supplement*, (91):146–159, 1987.

[68] W.H. Press, S.A Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in 'C': The Art of Scientific Computing*, chapter 10, page 337ff. Cambridge University Press, second edition, 1992.

[69] R.V. Reid. *Ann. Physics*, 50:411, 1968.

[70] J.M. Irvine. *Nuclear Structure Theory.* Pergamon Press, 1972.

[71] Y. Fujiwara and K. T. Hecht. RGM Study of the NN Interaction in an Extended Quark Model. *Nuclear Physics*, A462:621–660, 1987.

[72] I.S. Gradshteyn. *Table of Integrals, Series and Products.* Orlando London Academic Press, 1980.

[73] Milton Abramowitz and Irene A. Stegun, editors. *Handbook of Mathematical Functions.* Dover Publications, 1954.